

STOCHASTIC SELECTION PROCESSES

ALEX MCAVOY

ABSTRACT. We propose a mathematical framework for natural selection in finite populations. Traditionally, many of the selection-based processes used to describe cultural and genetic evolution (such as imitation and birth-death models) have been studied on a case-by-case basis. Over time, these models have grown in sophistication to include population structure, differing phenotypes, and various forms of interaction asymmetry, among other features. Furthermore, many processes inspired by natural selection, such as evolutionary algorithms in computer science, possess characteristics that should fall within the realm of a “selection process,” but so far there is no overarching theory encompassing these evolutionary processes. The framework of *stochastic selection processes* we present here provides such a theory and consists of three main components: a *population state space*, an *aggregate payoff function*, and an *update rule*. A population state space is a generalization of the notion of population structure, and it can include non-spatial information such as strategy-mutation rates and phenotypes. An aggregate payoff function allows one to generically talk about the fitness of traits without explicitly specifying a method of payoff accounting or even the nature of the interactions that determine payoff/fitness. An update rule is a fitness-based function that updates a population based on its current state, and it includes as special cases the classical update mechanisms (Moran, Wright-Fisher, etc.) as well as more complicated mechanisms involving chromosomal crossover, mutation, and even complex cultural syntheses of strategies of neighboring individuals. Our framework covers models with variable population size as well as with arbitrary, measurable trait spaces.

1. INTRODUCTION

Evolutionary game theory has proven itself extremely useful for modeling both cultural and genetic evolution (Maynard Smith, 1982; Hofbauer and Sigmund, 1998; Dugatkin, 2000; Nowak, 2006). Traits, which are represented as strategies, determine the fitness of the players. An individual’s fitness might be determined solely by his or her strategy (frequency-independent fitness) or it might also depend on the traits of the other players in the population (frequency-dependent fitness). The population evolves via a fitness-based update mechanism, and the long-run behavior of this process can be studied to determine which traits are more successful than others.

Evolutionary game theory was first used to study evolution in infinite populations via deterministic replicator dynamics (Taylor and Jonker, 1978). More recently, the dynamics of evolutionary games have also been studied in finite populations (Nowak et al., 2004; Taylor et al., 2004). Finite-population evolutionary dynamics typically have two timescales: one for interactions and one for updates. A player has a strategy (trait) and interacts with his or her neighbors in order to receive a payoff. In a birth-death process, for instance, this payoff is converted to reproductive fitness and used to update the population as follows: First, a player is chosen from the population for reproduction with probability proportional to (relative) fitness. Next, a player is chosen uniformly at random from the population for death, and the offspring of the reproducing player replaces the deceased player. This birth-death process is a frequency-dependent version of the classical Moran process (Moran, 1958; Nowak, 2006).

Whereas replicator dynamics are deterministic, finite-population models of evolution are inherently stochastic and incorporate principles of both natural selection and genetic drift. All biological populations are finite, and we focus here on stochastic evolutionary games in finite populations. In particular, we focus on *selection* processes, which, informally, are evolutionary processes in which the update step depends on fitness. Selection processes typically resemble birth-death processes in that there is reproduction and replacement, with “fitter” players more likely to reproduce than those with lower fitness. Of course, the order and number of births and deaths may vary, and the update might instead be based on imitation instead of reproduction. One of our goals here is to precisely define *selection process* in a way that captures all of the salient features of the classical models of evolution in finite populations.

Nowak et al. (2009) state that “There is (as yet) no general mathematical framework that would encompass evolutionary dynamics for any kind of population structure.” We propose here such a framework, which we term *stochastic selection processes*, to describe the evolutionary games used to model natural selection in finite populations. Stochastic selection processes model evolutionary processes with two timescales: one for interactions (which determine fitness) and one for updates (selection, mutation, etc.). Our framework takes into account arbitrary population structures, as well as non-spatial information about the population such as phenotypes and strategy mutations. Moreover, this framework encompasses all types of strategy spaces, games (matrix, asymmetric, multiplayer, etc.), and fitness-based update rules.

An example of ambiguity in evolutionary game theory is that classical games, such as two-player matrix games, are often used to define evolutionary processes in populations, and in this context the term “game” can refer to either the classical game or to the evolutionary process. Moreover, classical multiplayer games, such as the public goods game, can result in processes in which each player in the population derives a payoff from *several* multiplayer interactions, and some of these interactions might involve players who are not neighbors (see McAvoy and Hauert, 2015a). Even further, when a player is involved in multiple interactions, the total payoff to this player may be derived in more than one way: payoffs from individual encounters may be accumulated (added) or averaged, for instance, and the nature of this accounting can strongly influence the evolutionary process (Maciejewski et al., 2014).

In order to accommodate the many ways of deriving “total payoff” from a classical game, one may distill from these methods a common feature: each player has a strategy and receives an aggregate payoff from a sequence of interactions. That is, if S is the strategy space available to each of the N players in the population, then there is an *aggregate payoff function*, $u : S^N \rightarrow \mathbb{R}^N$, such that the i th coordinate function, u_i , is the payoff to player i for all of the (microscopic) interactions in which this player is involved. As an example, consider the two-player game defined by the matrix

$$\begin{array}{cc} & \begin{matrix} A & B \end{matrix} \\ \begin{matrix} A \\ B \end{matrix} & \begin{pmatrix} a & b \\ c & d \end{pmatrix} \end{array}. \quad (1)$$

Suppose that the population is well mixed so that each player interacts with every other player. Let $S = \{A, B\}$ and let $s \in S^N$ be a strategy profile consisting of $k + 1$ players using A and $N - 1 - k$ players using B (the ordering is not important since the population is well mixed). If player i is an A -player, then the payoff to this player is

$$u_i^{\text{acc}}(s) := ka + (N - 1 - k)b \quad (2)$$

if payoffs are accumulated and

$$u_i^{\text{ave}}(s) := \frac{ka + (N - 1 - k)b}{N - 1} \quad (3)$$

if payoffs are averaged. These two methods of calculating payoffs from pairwise interactions give essentially equivalent evolutionary dynamics since the population is well mixed, but this phenomenon need not hold for more complicated methods of payoff accounting or (spatial) population structures. Evolutionary dynamics aside, this example illustrates how one can define an aggregate payoff function, u , from a classical game such as a 2×2 matrix game; different methods of obtaining total payoff from a series of interactions result in different aggregate payoff functions.

A selection process in a finite population typically has for a state space the set of all strategy profiles, i.e. the set of all N -tuples of strategies (Allen and Tarnita, 2012). A strategy profile indicates a strategy for each player in the population, and often an evolutionary process updates only these strategies. The population structure may be fixed (Lieberman et al., 2005; Szabó and Fáth, 2007) or dynamic (Tarnita et al., 2009; Wardil and Hauert, 2014). (In the latter case, one must also account for the population structure in the state space of the process.) An aggregate payoff function, which takes into account both strategies and population structure, assigns a payoff to each player in the population. The payoff to player i , u_i , is then converted to fitness, $f_i = f(u_i)$, where f is some payoff-to-fitness function (e.g. $f(u_i) = \exp\{\beta u_i\}$ for some $\beta \geq 0$). A fitness-based update rule such as birth-death (Moran, 1958; Nowak et al., 2004); death-birth or imitation (Ohtsuki et al., 2006; Ohtsuki and Nowak, 2006); pairwise comparison (Szabó and Tóke, 1998;

Traulsen et al., 2007); or Wright-Fisher (Ewens, 2004; Imhof and Nowak, 2006) is then repeatedly applied to the population, at each step updating the strategies of the players and (possibly) the population structure.

Based on this pattern, it seems reasonable to define the state of an evolutionary process to be a pair, (s, Γ) , where s is an N -tuple of strategies and Γ is a population structure. However, there may be more to the state of a population than its spatial structure. For example, Antal et al. (2009) consider evolution in *phenotype space*, a model in which each player has both a strategy and a phenotype, and phenotypes influence the effects of strategies on interactions. In the Prisoner’s Dilemma, for instance, a cooperator (whose strategy is C) cooperates only with other players who are phenotypically similar and defects otherwise. In this instance, we show how one can consider phenotypes as a part of the “population state” in the sense that they contain information about the players in the population. In general, a state of the evolutionary process can be represented by a pair, (s, \mathcal{P}) , where s is a strategy profile and \mathcal{P} is a *population state* (which we make mathematically precise). Notably, the population state is distinct from the strategies of the players; it describes all of the non-strategy information about the players.

When viewed from this perspective, the effects of phenotypes on strategies can be implemented directly in the aggregate payoff function, u : a player facing a cooperator receives the payoff for facing a cooperator if and only if they are phenotypically similar. Similarly, strategy mutations can also be considered as a part of the population state and accounted for directly in the update step of the process. As a part of our analysis, we formally define *aggregate payoff function* and *update rule* and show how they are influenced by the components of the population state.

This setup, which involves a series of interactions as the population transitions through various states, is reminiscent of a *stochastic game* (Shapley, 1953). A stochastic game is played in stages, with each stage consisting of a normal-form game determined by some “state.” The game played in the subsequent stage is determined probabilistically by the current state as well as the strategies played in the current stage. We will see that, in general, selection processes are not necessarily stochastic games, and neither are stochastic games necessarily selection processes. However, these two types of processes do share some common features, and we use some of the components of a stochastic game as inspiration for our framework.

Many problems in evolution, such as how and why cooperation evolves, depend on the specifics of the update rule, population structure, mutation rates, etc. (Ohtsuki et al., 2006; Taylor et al., 2007; Traulsen et al., 2009; Débarre et al., 2014; Rand et al., 2014). We clarify how these pieces (among others) fit together. Our objective here is threefold: (1) to compare and contrast evolutionary and stochastic games, drawing inspiration from the latter to describe the former; (2) to propose a general mathematical framework encompassing natural selection models in finite populations; and (3) to examine the components of several classical evolutionary processes and demonstrate how they fit into our framework. Many (if not most) of the existing models of evolution in finite populations involve a fixed population size. Therefore, our framework is first stated in terms of a fixed population size since this setting most readily allows for a comparison to the theory of stochastic games and for illustrative examples placing several standard evolutionary processes into a broader context. However, the assumption that the population size is fixed is not crucial to our theory, and we conclude by extending our framework to processes with variable population size.

2. STOCHASTIC GAMES

Prior to outlining the basic theory of stochastic games, we first need to recall some definitions and notation. A *measurable space* consists of a set, X , and a σ -algebra of sets, $\mathcal{F}(X)$, on X . Often, we refer to X itself as a “measurable space” and suppress $\mathcal{F}(X)$. If X is a measurable space, then we denote by $\Delta(X)$ the space of probability measures on X ; that is, if $\mathcal{M}(X)$ is the space of all measures on $(X, \mathcal{F}(X))$, then

$$\Delta(X) := \{\mu \in \mathcal{M}(X) : \mu(X) = 1\}. \quad (4)$$

For measurable spaces X and Y , denote by $K(X, Y)$ the set of Markov kernels from X to Y ; that is, $K(X, Y)$ is the set of functions $\kappa : X \times \mathcal{F}(Y) \rightarrow [0, 1]$ such that (i) $\kappa(x, -) : \mathcal{F}(Y) \rightarrow [0, 1]$ is in $\Delta(Y)$ for each $x \in X$ and (ii) $\kappa(-, E) : X \rightarrow [0, 1]$ is measurable for each $E \in \mathcal{F}(Y)$. We also write $\kappa : X \rightarrow \Delta(Y)$ to denote such a kernel.

Shapley (1953) considers a collection of normal-form games, together with a probabilistic rule for transitioning between these games, which he refers to as a *stochastic game*. A stochastic game is a generalization of a repeated game that, formally, consists of the following components:

- (i) N players, labeled $1, \dots, N$;
- (ii) for each player, i , a measurable strategy space, S_i ;
- (iii) a measurable state space, \mathcal{P} ;
- (iv) a “single-period” payoff function, $u : \mathbf{S} \times \mathcal{P} \rightarrow \mathbb{R}^N$, where u_i is the payoff to player i and $\mathbf{S} := S_1 \times \dots \times S_N$ is the set of all *strategy profiles*;
- (v) a transition kernel, $T : \mathbf{S} \times \mathcal{P} \rightarrow \Delta(\mathcal{P})$.

A Markov decision process is a stochastic game with one player (Puterman, 1994; Neyman, 2003), and a repeated game is a stochastic game whose state space, \mathcal{P} , consists of just a single element (McMillan, 2001; Mertens et al., 2015).

Examples of strategies for stochastic games are the following (see Neyman and Sorin, 2003):

- (1) *Pure strategies*: Let \mathcal{H} denote the set of all possible histories, i.e.

$$\mathcal{H} := \{\emptyset\} \cup \bigsqcup_{t \geq 1} (\mathbf{S} \times \mathcal{P})^t, \quad (5)$$

where \emptyset denotes the “null” history and $\bigsqcup_{t \geq 1} (\mathbf{S} \times \mathcal{P})^t$ is the disjoint union of the spaces of t -tuples, $(\mathbf{S} \times \mathcal{P})^t$, for $t \geq 1$. A pure strategy for player i is a map

$$s_i : \mathcal{H} \longrightarrow S_i, \quad (6)$$

indicating an action in S_i for each t and history $h^t \in (\mathbf{S} \times \mathcal{P})^t \subseteq \mathcal{H}$. We denote by $\text{Map}(\mathcal{H}, S_i)$ the set of all such maps, i.e. the set of player i ’s pure strategies.

- (2) *Mixed strategies*: A mixed strategy for player i is a probability distribution over the set of pure strategies for player i , i.e. an element $\sigma_i \in \Delta(\text{Map}(\mathcal{H}, S_i))$.
- (3) *Behavioral strategies*: A behavioral strategy for player i is a map

$$\sigma_i : \mathcal{H} \longrightarrow \Delta(S_i), \quad (7)$$

indicating a distribution over S_i for each t and history $h^t \in (\mathbf{S} \times \mathcal{P})^t \subseteq \mathcal{H}$.

- (4) *Markov strategies*: A Markov strategy for player i is a behavioral strategy, σ_i , such that $\sigma_i(h^t) = \sigma_i(k^t)$ for each t and $h^t, k^t \in (\mathbf{S} \times \mathcal{P})^t \subseteq \mathcal{H}$ with $h_t^t = k_t^t$. In other words, a Markov strategy is a “memory-one” behavioral strategy, i.e. a behavioral strategy that depends on only the last strategy profile, state, and t .
- (5) *Stationary strategies*: A stationary strategy is a Markov strategy that is independent of t , i.e. a behavioral strategy that depends on only the last strategy profile and state.

Of these five classes of strategies, behavioral strategies are the most general. Indeed, pure, mixed, Markov, and stationary strategies are all instances of behavioral strategies. In the context of repeated games, a memory-one strategy of the repeated game (Press and Dyson, 2012) is equivalent to a stationary strategy of the stochastic game, and a longer-memory strategy of the repeated game (Hauert and Schuster, 1997) is equivalent to a behavioral strategy.

2.1. Evolutionary processes as stochastic games. At first glance, stochastic games seem to provide a reasonable framework for evolutionary games: a stochastic game transitions through states in stages (“periods”), which could be population structures or states, and in each stage the players receive payoffs based on a single-period payoff function, u . However, it is in the dynamics that the differences between stochastic and evolutionary games become evident:

The combination of a stochastic game and a strategy for each player defines a stochastic process on $\mathbf{S} \times \mathcal{P}$, although this stochastic process might or might not be a Markov chain. For example, let T be the transition kernel for a stochastic game, and suppose that σ_i is a stationary strategy for player i for $i = 1, \dots, N$. Let κ be the transition kernel on $\mathbf{S} \times \mathcal{P}$ defined by the product measure, i.e.

$$\begin{aligned} \kappa : \mathbf{S} \times \mathcal{P} &\longrightarrow \Delta(\mathbf{S} \times \mathcal{P}) \\ (s, \mathcal{P}) &\longmapsto \sigma_1[s, \mathcal{P}] \times \dots \times \sigma_N[s, \mathcal{P}] \times T[s, \mathcal{P}]. \end{aligned} \quad (8)$$

Thus, the stochastic game—together with this profile of stationary strategies—defines a time-homogeneous Markov chain on $\mathbf{S} \times \mathcal{P}$. In general, if these strategies are instead Markov strategies, then the resulting

Markov chain might be time-inhomogeneous. If these strategies are pure, mixed, or behavioral, then the stochastic process on $\mathbf{S} \times \mathcal{P}$ defined by the game need not even be a Markov chain.

Evolutionary processes are typically defined as Markov chains on $\mathbf{S} \times \mathcal{P}$, where \mathcal{P} is chosen to be a state space appropriate for the evolutionary process (such as the space of population structures, mutation-rate profiles, or phenotype profiles). In light of the previous remarks, it is not unreasonable to expect that many of these processes are equivalent to stochastic games combined with stationary strategies. As it turns out, evolutionary processes generally possess correlations between updates of strategies and population states, forbidding an equivalence between an evolutionary process and a Markov chain constructed from a stochastic game via Eq. (8). These correlations are evident already in one of the most basic models of evolution in finite populations:

Example 1 (Moran process). Suppose that $S = \{A, B\}$, where strategy A represents the mutant type and strategy B represents the wild type. A mutant type has fitness $r > 0$ relative to the wild type (whose fitness relative to itself is just 1). Let \mathbf{m} be a finite subset of $[0, 1]$ consisting of a number of “mutation rates,” and let $\mathcal{P} := \mathbf{m}^N$. In a well-mixed population of size N , the Moran process proceeds as follows: In each time step, an individual (“player”) is chosen for reproduction with probability proportional to relative fitness. Another player (including the one chosen for reproduction) is then chosen uniformly at random from the population for death. The offspring of the player chosen for reproduction replaces the deceased player. If player i is chosen for reproduction and $\varepsilon_i \in \mathbf{m}$ is this player’s mutation rate, then the offspring of this player inherits the type of the parent with probability $1 - \varepsilon_i$ and takes on a type uniformly at random from $\{A, B\}$ with probability ε_i . The mutation rate of the offspring is ε_i , which is inherited directly from the parent. Thus, a state of this process consists of a profile of types (“strategies”), $s \in S^N$, and a profile of mutation rates, $\varepsilon \in \mathcal{P} = \mathbf{m}^N$.

A transition between states (s, ε) and (s', ε') is possible only if there exists j such that $s_\ell = s'_\ell$ and $\varepsilon_\ell = \varepsilon'_\ell$ for each $\ell \neq j$. If player i is selected for reproduction and player j is chosen for death, then it must be the case that $\varepsilon_i = \varepsilon'_j$. If $\delta_{s,t} = 1$ when $s = t$ (and is 0 otherwise), then the probability that player i is selected for reproduction is

$$\frac{r\delta_{s_i,A} + \delta_{s_i,B}}{\sum_{\ell=1}^N (r\delta_{s_\ell,A} + \delta_{s_\ell,B})}. \quad (9)$$

The probability that player j is chosen for death is $1/N$. If the offspring of player i inherits the strategy of the parent, then it must be true that $s_i = s'_j$. Otherwise, the offspring of player i “mutates” and adopts strategy s'_j with probability $1/2$. Therefore, the probability of transitioning between states (s, ε) and (s', ε') is

$$\begin{aligned} \mathbf{T}_{(s,\varepsilon),(s',\varepsilon')} &= \sum_{i,j=1}^N \left(\prod_{\ell \neq j} \delta_{s_\ell, s'_\ell} \delta_{\varepsilon_\ell, \varepsilon'_\ell} \right) \delta_{\varepsilon_i, \varepsilon'_j} \\ &\quad \times \left(\frac{r\delta_{s_i,A} + \delta_{s_i,B}}{\sum_{\ell=1}^N (r\delta_{s_\ell,A} + \delta_{s_\ell,B})} \right) \left(\frac{1}{N} \right) \left[(1 - \varepsilon_i) \delta_{s_i, s'_j} + \varepsilon_i \left(\frac{1}{2} \right) \right]. \end{aligned} \quad (10)$$

By Eq. (10), the distributions on S^N and \mathcal{P} , respectively, are *not* independent.

More formally, consider a Markov chain on $\mathbf{S} \times \mathcal{P}$ defined by some evolutionary process such as the Moran process of Example 1, and let κ be its transition kernel. The projection maps $\Pi_1 : \mathbf{S} \times \mathcal{P} \rightarrow \mathbf{S}$ and $\Pi_2 : \mathbf{S} \times \mathcal{P} \rightarrow \mathcal{P}$ produce pushforward maps

$$\begin{aligned} (\Pi_1)_* : \Delta(\mathbf{S} \times \mathcal{P}) &\longrightarrow \Delta(\mathbf{S}) \\ &: \mu \longmapsto \mu \circ \Pi_1^{-1}; \end{aligned} \quad (11a)$$

$$\begin{aligned} (\Pi_2)_* : \Delta(\mathbf{S} \times \mathcal{P}) &\longrightarrow \Delta(\mathcal{P}) \\ &: \mu \longmapsto \mu \circ \Pi_2^{-1}. \end{aligned} \quad (11b)$$

From κ , we obtain a transition kernel for a stochastic game, T , defined by

$$T[s, \mathcal{P}] := (\Pi_2)_* \kappa[s, \mathcal{P}] \quad (12)$$

for each $s \in \mathbf{S}$ and $\mathcal{P} \in \mathcal{P}$. Similarly, we obtain the (stationary) strategy profile

$$\sigma[s, \mathcal{P}] := (\Pi_1)_* \kappa[s, \mathcal{P}]. \quad (13)$$

However, one typically loses information in passing from such a Markov chain to the combination of a stochastic game and a profile of stationary strategies. First of all, the transition kernel κ generally cannot be reconstructed from T and σ since $\kappa[s, \mathcal{P}]$ need not be in $\Delta(\mathbf{S}) \times \Delta(\mathcal{P})$; a priori, we know only that $\kappa[s, \mathcal{P}] \in \Delta(\mathbf{S} \times \mathcal{P})$ for $s \in \mathbf{S}$ and $\mathcal{P} \in \mathcal{P}$ (see Eq. (10), for example). Moreover, σ need not be of the form $(\sigma_1, \dots, \sigma_N)$ for stationary strategies $\sigma_1, \dots, \sigma_N$; in particular, σ is a *correlated* stationary profile (see Aumann, 1987; Fudenberg and Tirole, 1991). In other words, whereas a sequence of independent strategy choices produce an element

$$(\sigma_1[s, \mathcal{P}], \dots, \sigma_N[s, \mathcal{P}]) \in \Delta(S_1) \times \dots \times \Delta(S_N), \quad (14)$$

it might be the case that $\sigma[s, \mathcal{P}] \in \Delta(S_1 \times \dots \times S_N) - (\Delta(S_1) \times \dots \times \Delta(S_N))$.

In §3, we present a framework for stochastic evolutionary processes used to model natural selection. These processes, which we call *stochastic selection processes*, illustrate more clearly the correlations between distributions on \mathbf{S} and \mathcal{P} arising in many evolutionary processes. We saw in Example 1 that, despite the similarities between stochastic games and evolutionary processes, there are important differences between the two frameworks. However, our notion of a stochastic selection process draws inspiration from the theory of stochastic games. Namely, we appropriate the concepts of (i) state space, \mathcal{P} ; (ii) single-period payoff function, u ; and (iii) update step.

3. STOCHASTIC SELECTION PROCESSES WITH FIXED POPULATION SIZE

Here we focus on a type of evolutionary process that we term a *stochastic selection process*. Stochastic selection processes seek to model processes with two timescales: one for interactions and one for selection. In the interaction step, players interact with one another and receive payoffs based on their strategies. In the selection step, the population is updated probabilistically based on the current population and the players' payoffs. Selection processes provide a general framework for the evolutionary games used to model processes based on natural selection (Maynard Smith, 1982).

Roughly speaking, the processes modeled by evolutionary game theory may be split into two classes: cultural and genetic (McAvoy and Hauert, 2015c). In cultural processes, there is a fixed set of players who repeatedly revise their strategies based on some update rule (such as imitation). In genetic processes, strategies are updated via reproduction and genetic inheritance. Naturally, a process need not be one or the other; there may be both cultural and genetic components in an evolutionary process. Unlike purely cultural processes, those with a genetic component have the property that the players themselves, as well as the size of the population, may actually change via births and deaths, thus it does not make sense to speak of a fixed population of players as in requirement (i) of a stochastic game. However, one can choose an arbitrary enumeration of the players in each step of the process and refer to the player labeled i at time n as “player i .” Of course, player i at time m and player i at time $n \neq m$ might be different, but a stochastic selection process is a Markov chain and the transition kernel can be defined for any enumeration of the players at a given time. The implicit property that natural selection does not depend on the enumeration of the players must be stated formally in terms of the update rule. Informally, the update rule for a stochastic selection process must satisfy a symmetry condition that guarantees the dynamics do not depend on these enumerations.

As an example, an evolutionary process based on a game with two-strategies in a well-mixed population may be modeled as a Markov chain whose state space is $\{0, 1, \dots, N\}$ (Nowak, 2006). If $S = \{A, B\}$ is the strategy set, then the state of the population is determined by the number of A -players, which is just an integer $i \in \{0, 1, \dots, N\}$. Alternatively, one may choose an arbitrary enumeration of the players and represent the state of the population as an element $(s_1, \dots, s_N) \in S^N$. Since the population is well mixed, evolutionary dynamics depend on only the frequency of each strategy in the population, so any two states (s_1, \dots, s_N) and (s'_1, \dots, s'_N) consisting of the same number of A -players should be indistinguishable. In other words, if \mathbf{T} is the transition matrix for an evolutionary game in this population, then $\mathbf{T}_{\pi s, \tau s'} = \mathbf{T}_{s, s'}$

for each $s, s' \in S^N$ and $\pi, \tau \in \mathfrak{S}_N$, where \mathfrak{S}_N acts on S^N by permuting the coordinates. Thus, the Markov chain is more naturally defined on the *quotient space*

$$\mathcal{S} := S^N / \mathfrak{S}_N, \quad (15)$$

which is isomorphic to $\{0, 1, \dots, N\}$ when $S = \{A, B\}$. (Recall that an action of a group, G , on a set, X , gives an equivalence relation, \sim , on X , which is defined by $x \sim x'$ if and only if there exists $g \in G$ with $x' = gx$. The quotient space, X/G , is defined as the set of equivalence classes under \sim , and the class containing x is denoted by “ $x \bmod G$.”) Of course, one may consider strategy spaces with more than two strategies: Suppose that $S = \{A_1, \dots, A_n\}$, and, for each $r = 1, \dots, n$, let $\psi_r : S^N \rightarrow \{0, 1, \dots, N\}$ be the map sending a strategy profile, $s \in S^N$, to the number of players using strategy A_r in s . Since $\varphi_r(\pi s) = \varphi_r(s)$ for each $\pi \in \mathfrak{S}_N$ and $s \in S^N$, the map

$$\begin{aligned} \Psi : S^N &\longrightarrow \left\{ (k_1, \dots, k_n) \in \{0, 1, \dots, N\}^n : k_1 + \dots + k_n = N \right\} \\ &: s \longmapsto \left(\psi_1(s), \dots, \psi_n(s) \right) \end{aligned} \quad (16)$$

descends to an isomorphism

$$\tilde{\Psi} : S^N / \mathfrak{S}_N \longrightarrow \left\{ (k_1, \dots, k_n) \in \{0, 1, \dots, N\}^n : k_1 + \dots + k_n = N \right\}. \quad (17)$$

Since an evolutionary update rule may be defined on the space of strategy-frequency profiles,

$$\left\{ (k_1, \dots, k_n) \in \{0, 1, \dots, N\}^n : k_1 + \dots + k_n = N \right\}, \quad (18)$$

we see once again that the Markov chain defined by an evolutionary process in this population naturally has S^N / \mathfrak{S}_N for a state space. We show here that this phenomenon generalizes to arbitrary types of populations and update rules. In the process of establishing this general construction, we must formally define *population state* (§3.1) and *update rule* (§3.3).

We first assume that the population size, N , is fixed. This assumption allows us to place many of the classical (fixed population size) stochastic evolutionary processes into the context of our framework. After discussing the components of a selection process and giving several examples, we formally define *stochastic selection process* in its full generality (covering populations of variable size) in §4.

Just like a stochastic game, a stochastic selection process consists of a measurable state space, \mathcal{P} , and a strategy space for each player. We assume that

$$S_1 = S_2 = \dots = S_N =: S, \quad (19)$$

so that $\mathbf{S} = S^N$. This assumption that the players all have the same strategy space is not restrictive since the dynamics of the process define the evolution of strategies; one can just enlarge each player’s strategy space if necessary and let the dynamics ensure that those strategies that are not available to a given player are never used. Before discussing these dynamics, we first need to explore the state space, \mathcal{P} :

3.1. Population states. We seek to appropriate the idea of a state space, \mathcal{P} , of a stochastic game in order to introduce population structure into an evolutionary process. In fact, a population’s spatial structure (such as a graph) is just one component of this space; mutation rates or phenotypes may also be included in this space. Therefore, rather than declaring \mathcal{P} to be a space of population *structures*, we say that \mathcal{P} is the space of population *states*. A population state indicates properties of the players and relationships between the players. (Note that “population state” in this context does not include the strategies of the players in the population.) If one enumerates these players differently, then there should be a corresponding “relabeling” of the population state so that these properties and relationships are preserved. Since changing the enumeration of the population amounts to applying an element of \mathfrak{S}_N (the symmetric group on N letters) to $\{1, \dots, N\}$, it follows that \mathcal{P} must be equipped with a group action of \mathfrak{S}_N . Thus, if $s \in S^N$ is a strategy profile of the population and $\mathcal{P} \in \mathcal{P}$ is a population state, then the pair (s, \mathcal{P}) represents the same population of players as $(\pi s, \pi \mathcal{P})$ whenever $\pi \in \mathfrak{S}_N$. In other words, the population state space, $\mathcal{P} = \mathcal{P}_N$, which we write with a subscript to indicate the population size, is a measurable \mathfrak{S}_N -space. More formally:

Definition 1. A *population state space* for a set of N players is a measurable space, \mathcal{P}_N , equipped with an action of \mathfrak{S}_N in such a way that the map $\pi : \mathcal{P}_N \rightarrow \mathcal{P}_N$ is measurable for each $\pi \in \mathfrak{S}_N$. If \mathcal{P}_N is a population state space for a set of N players, then a *population state* for these players is simply an element $\mathcal{P} \in \mathcal{P}_N$.

3.1.1. Examples.

Example 2 (Graphs). Consider the set of $N \times N$, nonnegative matrices over \mathbb{R} ,

$$\mathcal{P}_N^G := \{ \Gamma \in \mathbb{R}^{N \times N} : \Gamma_{ij} \geq 0 \text{ for each } i, j = 1, \dots, N \}, \quad (20)$$

equipped with an action of \mathfrak{S}_N defined by $(\pi\Gamma)_{ij} = \Gamma_{\pi(i)\pi(j)}$. An element $\Gamma \in \mathcal{P}_N^G$ defines a directed, weighted graph whose vertices are $\{1, \dots, N\}$, with an edge from i to j if and only if $\Gamma_{ij} \neq 0$ (the weight of the edge is then Γ_{ij}). Γ is *undirected* if $\Gamma_{ij} = \Gamma_{ji}$ for each i and j , and Γ is *unweighted* if $\Gamma \in \{0, 1\}^{N \times N}$.

Example 3 (Sets). A set-structured population consists of a finite number of sets, each containing some subset of the population, such that each player is in at least one set (Tarnita et al., 2009). Set-structured populations may be modeled using relations:

$$\mathcal{P}_N^S := \{ R \subseteq \{1, \dots, N\} \times \{1, \dots, N\} : R \text{ is reflexive and symmetric} \}. \quad (21)$$

That is, if $R \in \mathcal{P}_N^S$, then $(i, i) \in R$ for each i (“reflexive”) and $(i, j) \in R$ if and only if $(j, i) \in R$ (“symmetric”). $R \in \mathcal{P}_N^S$ defines a set-structure with i and j in a common set if and only if $(i, j) \in R$. There is a natural action of \mathfrak{S}_N on \mathcal{P}_N^S defined by

$$(i, j) \in \pi R \iff (\pi i, \pi j) \in R, \quad (22)$$

which makes \mathcal{P}_N^S into a population state space.

Example 4 (Demes). A deme-structure on a population is a subdivision of the population into subpopulations, or “demes” (Taylor et al., 2001; Wakeley and Takahashi, 2004; Hauert and Imhof, 2012). Similar to set-structured populations, deme-structured populations may be modeled using relations (but with the stronger notion of *equivalence relation*):

$$\mathcal{P}_N^D := \{ R \subseteq \{1, \dots, N\} \times \{1, \dots, N\} : R \text{ is reflexive, symmetric, and transitive} \}. \quad (23)$$

$\mathcal{P}_N^D \subseteq \mathcal{P}_N^S$, and whereas a set-structured population may have overlapping sets, a deme-structured population has disjoint sets. The additional transitivity requirement guarantees that these sets partition $\{1, \dots, N\}$. The action of \mathfrak{S}_N on \mathcal{P}_N^D is the one inherited from \mathcal{P}_N^S , making \mathcal{P}_N^D into a population state space.

So far, we have considered population structures that describe spatial and qualitative relationships between the players. One could also associate to the players quantities such as *mutation rates* or *phenotypes*:

Example 5 (Mutation rates). Consider a process in which updates are based on births and deaths (such as a Moran or Wright-Fisher process). Moreover, suppose that the spatial structure of the population is a graph. If player i reproduces, then with probability ε_i the offspring adopts a novel strategy uniformly at random (“mutates”), and with probability $1 - \varepsilon_i$ the offspring inherits the strategy of the parent. The mutation rate, ε_i , is passed on directly from parent to offspring. The population state space for this process is then $\mathcal{P}_N := \mathcal{P}_N^G \times [0, 1]^N$; a population state consists of (i) a graph, indicating the spatial relationships between the players, and (ii) a profile of mutation rates, $\varepsilon \in [0, 1]^N$, with ε_i indicating the probability that the offspring of player i mutates. For update rules based on imitation, mutation rates might more appropriately be called “exploration rates,” and they are implemented slightly differently. In general, mutation rates appear in different forms and help to distinguish cultural and genetic update rules; we give several examples in §3.3. The upshot of this discussion is that a population state may consist of a spatial structure, such as a graph in \mathcal{P}_N^G , *as well* as some extra information pertaining to the players, such as mutation rates.

Example 6 (Phenotype space). In addition to strategies, the players may also have *phenotypes* that affect interactions with other players in the population (Antal et al., 2009; Nowak et al., 2009). If the spatial structure of the population is a graph and the phenotype of each player is a one-dimensional discrete quantity, then one may define the population state space to be $\mathcal{P}_N^G \times \mathbb{Z}^N$. Just as with mutation rates, the rest of the process determines how these phenotypes affect the dynamics. In §3.2, we continue this example and go

into the details of how the inclusion of phenotypes in the population state space affects the payoffs of the players, which can then be used to recover a model of evolution in *phenotype space* of Antal et al. (2009).

3.1.2. Symmetries of population states. The action of \mathfrak{S}_N on \mathcal{P}_N can be used to formally define a notion of population symmetry:

Definition 2 (Automorphism of population state). For a population state, \mathcal{P} , in a population state space, \mathcal{P}_N , an *automorphism* of \mathcal{P} is an element $\pi \in \mathfrak{S}_N$ such that $\pi\mathcal{P} = \mathcal{P}$. The group of automorphisms of \mathcal{P} is $\text{Aut}(\mathcal{P}) := \text{Stab}_{\mathfrak{S}_N}(\mathcal{P})$, where $\text{Stab}_{\mathfrak{S}_N}(\mathcal{P})$ denotes the stabilizer of \mathcal{P} under the group action of \mathfrak{S}_N on \mathcal{P}_N .

If $\Gamma \in \mathcal{P}_N^G$, for example, then $\text{Aut}(\Gamma)$ is the group of graph automorphisms of Γ in the classical sense, i.e. the set of $\pi \in \mathfrak{S}_N$ such that $\Gamma_{\pi(i)\pi(j)} = \Gamma_{ij}$ for each i and j . Such automorphisms have played an important role in the study of evolutionary games on graphs (see Taylor et al., 2007; Débarre et al., 2014). We discuss symmetries of population states further in §5.

Now that we have a formal definition of *population state space*, we explore how this space influences evolutionary dynamics. The processes we seek to model have two timescales: interactions and updates. In §3.2, we consider the influence of the population state on interactions, and in §3.3, we define *update rule* and show how the population state fits into the update step of an evolutionary process.

3.2. Aggregate payoff functions. Prior to stating the definition of a general payoff function for a stochastic selection process, we consider a motivating example that is based on a popular type of game used to model frequency-dependent fitness in evolutionary game theory:

Example 7. Consider the symmetric, two-player game whose payoff matrix is

$$\begin{array}{cc} & \begin{matrix} A_1 & A_2 \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \end{matrix} & \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}. \end{array} \quad (24)$$

If the population structure is a graph and the population state space is \mathcal{P}_N^G , then one can construct a function $u : \{1, 2\}^N \times \mathcal{P}_N^G \rightarrow \mathbb{R}^N$ by letting u_i be defined as

$$u_i\left((s_1, \dots, s_N), \Gamma\right) := \sum_{j=1}^N \Gamma_{ij} a_{s_i s_j}. \quad (25)$$

That is, u_i is the “aggregate payoff” function for player i since it produces the total payoff from player i ’s interactions with all of his or her neighbors, weighted appropriately by the edge weights of the population structure. If $\pi \in \mathfrak{S}_N$, then

$$\begin{aligned} u_i\left((s_{\pi(1)}, \dots, s_{\pi(N)}), \pi\Gamma\right) &= \sum_{j=1}^N \Gamma_{\pi(i)\pi(j)} a_{s_{\pi(i)} s_{\pi(j)}} \\ &= \sum_{j=1}^N \Gamma_{\pi(i)j} a_{s_{\pi(i)} s_j} \\ &= u_{\pi(i)}\left((s_1, \dots, s_N), \Gamma\right). \end{aligned} \quad (26)$$

Therefore, although $u^\Gamma := u(-, \Gamma) : \{1, 2\}^N \rightarrow \mathbb{R}^N$ need not be symmetric in the sense that $u^\Gamma(\pi s) = \pi u^\Gamma(s)$ for each $s \in \{1, 2\}^N$, it *is* symmetric in the sense that

$$u(\pi s, \pi\Gamma) = \pi u(s, \Gamma) \quad (27)$$

for each $s \in \{1, 2\}^N$ and $\Gamma \in \mathcal{P}_N = \mathcal{P}_N^G$. In other words, all of the information that results in payoff asymmetry is contained in the population state space, \mathcal{P}_N .

Using the function u and Eq. (27) of the previous example as motivation, we have:

Definition 3 (Aggregate payoff function). An *aggregate payoff function* is a map

$$u : S^N \times \mathcal{P}_N \longrightarrow \mathbb{R}^N \quad (28)$$

that satisfies $u(\pi s, \pi \mathcal{P}) = \pi u(s, \mathcal{P})$ for each $\pi \in \mathfrak{S}_N$, $s \in S^N$, and $\mathcal{P} \in \mathcal{P}_N$.

The symmetry condition in Definition 3, $u(\pi s, \pi \mathcal{P}) = \pi u(s, \mathcal{P})$, implies that an aggregate payoff function, u , is completely determined by the map $u_1 : S^N \times \mathcal{P}_N \rightarrow \mathbb{R}$. Indeed, if u_1 is known and $\pi \in \mathfrak{S}_N$ sends 1 to i , then $u_i(s, \mathcal{P}) = u_1(\pi s, \pi \mathcal{P})$ for each $s \in S^N$ and $\mathcal{P} \in \mathcal{P}_N$, which recovers the map $u : S^N \times \mathcal{P}_N \rightarrow \mathbb{R}^N$. This symmetry condition must hold even for individual encounters that are *asymmetric* (Example 8).

3.2.1. Examples.

Example 8. In place of (24), one could consider a collection of *bimatrices*,

$$\mathbf{M}^{ij} := \begin{matrix} & \begin{matrix} A_1 & A_2 \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \end{matrix} & \begin{pmatrix} a_{11}^{ij} & a_{12}^{ij} \\ a_{21}^{ij} & a_{22}^{ij} \end{pmatrix} \end{matrix}, \quad (29)$$

indexed by $i, j \in \{1, \dots, N\}$ (McAvoy and Hauert, 2015c). For each i and j , \mathbf{M}^{ij} is the payoff matrix for player i against player j , with the first coordinate of each entry denoting the payoff to player i and the second coordinate denoting the payoff to player j . The collection $\{\mathbf{M}^{ij}\}_{i,j=1}^N$ is equivalent to an element of $(\mathbb{R}^{2 \times 2})^{N \times N}$, i.e. a 2×2 real matrix indicating the payoff to player i against player j for each $i, j = 1, \dots, N$. In this case, the population state space consists of more than spatial structures; it also includes the details of the payoff-asymmetry appearing in individual encounters. In other words, if the population is graph-structured, then a population state consists of a graph and a collection of payoff matrices of the form of Eq. (29), i.e.

$$\mathcal{P}_N := \mathcal{P}_N^G \times (\mathbb{R}^{2 \times 2})^{N \times N}, \quad (30)$$

where the action of $\pi \in \mathfrak{S}_N$ on $(\mathbb{R}^{2 \times 2})^{N \times N}$ is $\pi(\mathbf{M}^{ij})_{i,j=1}^N := (\mathbf{M}^{\pi(i)\pi(j)})_{i,j=1}^N$. The aggregate payoff function, $u : S^N \times \mathcal{P}_N \rightarrow \mathbb{R}^N$, is defined by

$$u_i\left((s_1, \dots, s_N), \left(\Gamma, (\mathbf{M}^{ij})_{i,j=1}^N\right)\right) := \sum_{j=1}^N \Gamma_{ij} a_{s_i s_j}^{ij}. \quad (31)$$

For $\pi \in \mathfrak{S}_N$, we see that

$$\begin{aligned} u_i\left((s_{\pi(1)}, \dots, s_{\pi(N)}), \pi\left(\Gamma, (\mathbf{M}^{ij})_{i,j=1}^N\right)\right) &= \sum_{j=1}^N \Gamma_{\pi(i)\pi(j)} a_{s_{\pi(i)} s_{\pi(j)}}^{\pi(i)\pi(j)} \\ &= \sum_{j=1}^N \Gamma_{\pi(i)j} a_{s_{\pi(i)} s_j}^{\pi(i)j} \\ &= u_{\pi(i)}\left((s_1, \dots, s_N), \left(\Gamma, (\mathbf{M}^{ij})_{i,j=1}^N\right)\right), \end{aligned} \quad (32)$$

so u defines an aggregate payoff function in the sense of Definition 3.

Remark 1. For a fixed collection, $(\mathbf{M}^{ij})_{i,j=1}^N$, we could have instead let $\mathcal{P}_N = \mathcal{P}_N^G$ and

$$u_i\left((s_1, \dots, s_N), \Gamma\right) := \sum_{j=1}^N \Gamma_{ij} a_{s_i s_j}^{ij}. \quad (33)$$

However, for $\pi \in \mathfrak{S}_N$, we would then have

$$u_i\left((s_{\pi(1)}, \dots, s_{\pi(N)}), \pi\Gamma\right) = \sum_{j=1}^N \Gamma_{\pi(i)j} a_{s_{\pi(i)} s_j}^{i\pi^{-1}(j)}, \quad (34a)$$

$$u_{\pi(i)}\left((s_1, \dots, s_N), \Gamma\right) = \sum_{j=1}^N \Gamma_{\pi(i)j} a_{s_{\pi(i)} s_j}^{\pi(i)j}. \quad (34b)$$

The only way (34a) and (34b) are the same for each $s \in \{1, 2\}^N$ and $\Gamma \in \mathcal{P}_N^G$ is if $\mathbf{M}^{ij} = \mathbf{M}^{\pi(i)\pi(j)}$ for each $i, j = 1, \dots, N$, and this equality need not hold (which would mean that u , when defined in this way, is not an aggregate payoff function in the sense of Definition 3). Therefore, by enlarging \mathcal{P}_N via (30) and defining u via (31), we can essentially “factor out” the asymmetry present in the payoff function defined by (33). In other words, \mathcal{P}_N contains all of the non-strategy information that distinguishes the players’ payoffs.

Due to the separation of timescales in the selection processes we consider here, it often happens that u is independent of a portion of the population state space. More specifically, the population state space can be decomposed into an *interaction state space*, \mathcal{E}_N , and a *dispersal state space*, \mathcal{D}_N , such that $\mathcal{P}_N = \mathcal{E}_N \times \mathcal{D}_N$ and

$$u\left(s, (\mathcal{E}, \mathcal{D})\right) = u\left(s, (\mathcal{E}, \mathcal{D}')\right) \quad (35)$$

for each $s \in S^N$, $\mathcal{E} \in \mathcal{E}_N$, and $\mathcal{D}, \mathcal{D}' \in \mathcal{D}_N$. This decomposition generalizes models with separate interaction and dispersal *graphs* (Taylor et al., 2007; Ohtsuki et al., 2007a,b; Pacheco et al., 2009; Débarre et al., 2014). The interaction state, for instance, might consist of a population structure *and* other information (such as phenotypes):

Example 9 (Phenotype space, continued). Antal et al. (2009) study the evolution of cooperation in phenotype space. In terms of (24), each player has a strategy, A_1 (“cooperate”) or A_2 (“defect”), as well as a one-dimensional phenotype, which is simply an integer. If an interaction state has a graph as its spatial structure, then the interaction state space is $\mathcal{E}_N := \mathcal{P}_N^G \times \mathbb{Z}^N$. Thus, an interaction structure consists of a graph, $\Gamma \in \mathcal{P}_N^G$, and an N -tuple of phenotypes $r = (r_1, \dots, r_N) \in \mathbb{Z}^N$, where r_i is the phenotype of player i in the population. The phenotypes affect the strategies of the players as follows: cooperators cooperate with other neighbors with whom they share a phenotype, and they defect otherwise. Defectors always defect, regardless of phenotypic similarities. For each i , the payoff function, $u : \{1, 2\}^N \times \mathcal{E}_N \rightarrow \mathbb{R}^N$, satisfies

$$u_i\left((s_1, \dots, s_N), (\Gamma, r)\right) = \sum_{j=1}^N \Gamma_{ij} \left(\delta_{r_i, r_j} a_{s_i s_j} + (1 - \delta_{r_i, r_j}) a_{22} \right), \quad (36)$$

where δ_{r_i, r_j} is 1 if $r_i = r_j$ and 0 otherwise. Therefore, one can directly implement the influence of phenotype on strategy using the interaction state space, \mathcal{E}_N , and u .

3.3. Update rules. We saw at the beginning of §3 that for a game with n strategies in a well-mixed population, the state space for an evolutionary process is

$$S^N / \mathfrak{S}_N \cong \left\{ (k_1, \dots, k_n) \in \{0, 1, \dots, N\}^N : k_1 + \dots + k_n = N \right\}. \quad (37)$$

On the other hand, between update steps, one can simply fix some enumeration of the population and represent the state of the evolutionary process by an element of S^N , i.e. a *representative* of the state space, S^N / \mathfrak{S}_N . In populations with spatial structure, mutation rates, phenotypic differences, etc., this representative contains more information than simply a strategy profile; it also contains information about the population state. In other words, at a fixed point in time, the state of the evolutionary process can be described by an element of $S^N \times \mathcal{P}_N$, where \mathcal{P}_N is a population state space. Of course, the evolutionary dynamics of the process should not be affected by the choice of enumeration of the players in each time step, which means that the state space for the evolutionary process is naturally the quotient space

$$\mathcal{S} := (S^N \times \mathcal{P}_N) / \mathfrak{S}_N, \quad (38)$$

generalizing the state space of Eq. (37) to structured populations.

We now wish to describe the update step of an evolutionary process on \mathcal{S} . This update rule should not depend on how the players in the updated population are labeled. For example, if a player dies and is replaced by the offspring of another player, then the result of this death and replacement is a new element of \mathcal{S} . In other words, the new population does *not* lie in $S^N \times \mathcal{P}_N$ in a natural way; we must choose an enumeration of the players in order to get an element of $S^N \times \mathcal{P}_N$, and this enumeration may be arbitrary. Therefore,

given the current strategy profile and population state, an update rule should give a probability distribution over the state space of the process, \mathcal{S} (*not* $S^N \times \mathcal{P}_N$). On the other hand, in order to update the state of the population, one needs to speak of the likelihood that each player in the current state is updated. To do so, one may choose a representative of the current state of the process, $(s, \mathcal{P}) \in S^N \times \mathcal{P}_N$, which is equivalent to choosing a labeling of the players at that point in time. Again, the distribution over \mathcal{S} (conditioned on the current state of the process) should not depend on the labeling of the current state. Finally, this distribution over \mathcal{S} is a function of the *fitness profile* of the population; each player has a real-valued fitness, and the update rule depends on these values. Putting these components together, we have:

Definition 4 (Update rule). An *update rule* is a map,

$$\mathcal{U} : \mathbb{R}^N \longrightarrow K(S^N \times \mathcal{P}_N, \mathcal{S}), \quad (39)$$

that satisfies the symmetry condition

$$\mathcal{U}[\pi x] \left((\pi s, \pi \mathcal{P}), E \right) = \mathcal{U}[x] \left((s, \mathcal{P}), E \right) \quad (40)$$

for each $\pi \in \mathfrak{S}_N$, $x \in \mathbb{R}^N$, $(s, \mathcal{P}) \in S^N \times \mathcal{P}_N$, and $E \in \mathcal{F}(\mathcal{S})$, where $\mathcal{F}(\mathcal{S})$ is the quotient σ -algebra on $\mathcal{S} = (S^N \times \mathcal{P}_N) / \mathfrak{S}_N$ derived from $\mathcal{F}(S)$ and $\mathcal{F}(\mathcal{P}_N)$.

That is, an update rule is a family of Markov kernels,

$$\left\{ \mathcal{U}[x] \right\}_{x \in \mathbb{R}^N} \subseteq K(S^N \times \mathcal{P}_N, \mathcal{S}), \quad (41)$$

parametrized by the fitness profiles of the population, $x \in \mathbb{R}^N$, and satisfying Eq. (40). Eq. (40) says that the update does not depend on how the current population is represented. In other words, if (s, \mathcal{P}) and $(\pi s, \pi \mathcal{P})$ are two representatives of the same population at time t , then the update rule treats (s, \mathcal{P}) and $(\pi s, \pi \mathcal{P})$ as the same population. (If x is the fitness profile corresponding to the representative (s, \mathcal{P}) , then πx is the fitness profile corresponding to the representative $(\pi s, \pi \mathcal{P})$.)

Together, an update rule and aggregate payoff function define a Markov chain on \mathcal{S} whose kernel, κ , is constructed as follows: Let $f : \mathbb{R} \rightarrow \mathbb{R}$ be a *payoff-to-fitness* map, i.e. a function that converts a player's payoff to fitness. Consider the function

$$\begin{aligned} F : \mathbb{R}^N &\longrightarrow \mathbb{R}^N \\ &: (x_1, \dots, x_N) \longmapsto (f(x_1), \dots, f(x_N)), \end{aligned} \quad (42)$$

which converts payoff profiles to fitness profiles. If $u : S^N \times \mathcal{P}_N \rightarrow \mathbb{R}^N$ is an aggregate payoff function, then, for $(s, \mathcal{P}) \in S^N \times \mathcal{P}_N$ and $E \in \mathcal{F}(\mathcal{S})$, we let

$$\kappa \left((s, \mathcal{P}) \bmod \mathfrak{S}_N, E \right) := \mathcal{U} \left[F(u(s, \mathcal{P})) \right] \left((s, \mathcal{P}), E \right). \quad (43)$$

κ is well defined since, for each $\pi \in \mathfrak{S}_N$,

$$\begin{aligned} \kappa \left((\pi s, \pi \mathcal{P}) \bmod \mathfrak{S}_N, E \right) &= \mathcal{U} \left[F(u(\pi s, \pi \mathcal{P})) \right] \left((\pi s, \pi \mathcal{P}), E \right) \\ &= \mathcal{U} \left[\pi F(u(s, \mathcal{P})) \right] \left((\pi s, \pi \mathcal{P}), E \right) \\ &= \mathcal{U} \left[F(u(s, \mathcal{P})) \right] \left((s, \mathcal{P}), E \right) \\ &= \kappa \left((s, \mathcal{P}) \bmod \mathfrak{S}_N, E \right), \end{aligned} \quad (44)$$

where the second and third lines come from Eqs. (27) and (40), respectively.

3.3.1. Update pre-rules. Despite the fact that the evolutionary processes we seek to model here naturally have $\mathcal{S} = (S^N \times \mathcal{P}_N) / \mathfrak{S}_N$ for a state space, many evolutionary processes in the literature are defined directly on $S^N \times \mathcal{P}_N$ (see Allen and Tarnita, 2012). Update rules are sometimes cumbersome to write out explicitly, and defining a Markov chain on $S^N \times \mathcal{P}_N$ instead of on $(S^N \times \mathcal{P}_N) / \mathfrak{S}_N$ can simplify the presentation of the transition kernel. In this context, the notion of “update rule” still makes sense, but we instead call it an *update pre-rule* to distinguish it from the update rule of Definition 4:

Definition 5 (Update pre-rule). An *update pre-rule* is a map,

$$\mathcal{U}_0 : \mathbb{R}^N \longrightarrow K(S^N \times \mathcal{P}_N, S^N \times \mathcal{P}_N), \quad (45)$$

such that for each $\pi \in \mathfrak{S}_N$, $x \in \mathbb{R}^N$, $(s, \mathcal{P}) \in S^N \times \mathcal{P}_N$, and $E \in \mathcal{F}(\mathcal{S})$,

$$\mathcal{U}_0[\pi x] \left((\pi s, \pi \mathcal{P}), E \right) = \mathcal{U}_0[x] \left((s, \mathcal{P}), \tau E \right) \quad (46)$$

for some $\tau \in \mathfrak{S}_N$.

In many cases, the permutation τ is just π^{-1} , i.e.

$$\mathcal{U}_0[\pi x] \left((\pi s, \pi \mathcal{P}), \pi E \right) = \mathcal{U}_0[x] \left((s, \mathcal{P}), E \right) \quad (47)$$

for each $\pi \in \mathfrak{S}_N$. However, all that an evolutionary process on $S^N \times \mathcal{P}_N$ really requires is that if state $(\pi s, \pi \mathcal{P})$ is updated to (s', \mathcal{P}') , then state (s, \mathcal{P}) is updated to $(\tau s', \tau \mathcal{P}')$ for some τ . To relate Definitions 4 and 5, consider the projection map,

$$\begin{aligned} \Pi : S^N \times \mathcal{P}_N &\longrightarrow (S^N \times \mathcal{P}_N) / \mathfrak{S}_N = \mathcal{S} \\ &: (s, \mathcal{P}) \longmapsto (s, \mathcal{P}) \bmod \mathfrak{S}_N. \end{aligned} \quad (48)$$

Π gives rise to a pushforward map on measures,

$$\begin{aligned} \Pi_* : \Delta(S^N \times \mathcal{P}_N) &\longrightarrow \Delta(\mathcal{S}) \\ &: \mu \longmapsto \mu \circ \Pi^{-1}, \end{aligned} \quad (49)$$

which can be used to naturally derive an update rule from an update pre-rule:

Proposition 1. An update pre-rule canonically defines an update rule.

Proof. Let \mathcal{U}_0 be an update pre-rule and consider the map

$$\begin{aligned} \Pi_* \mathcal{U}_0 : \mathbb{R}^N &\longrightarrow K(S^N \times \mathcal{P}_N, \mathcal{S}) \\ &: x \longmapsto \left\{ (s, \mathcal{P}) \mapsto \Pi_* \mathcal{U}_0[x] \left((s, \mathcal{P}), - \right) \right\}. \end{aligned} \quad (50)$$

For $\pi \in \mathfrak{S}_N$, $x \in \mathbb{R}^N$, $(s, \mathcal{P}) \in S^N \times \mathcal{P}_N$, and $E \in \mathcal{F}(\mathcal{S})$, there exists $\tau \in \mathfrak{S}_N$ such that

$$\begin{aligned} (\Pi_* \mathcal{U}_0)[\pi x] \left((\pi s, \pi \mathcal{P}), E \right) &= \mathcal{U}_0[\pi x] \left((\pi s, \pi \mathcal{P}), \Pi^{-1} E \right) \\ &= \mathcal{U}_0[x] \left((s, \mathcal{P}), \tau \Pi^{-1} E \right) \\ &= \mathcal{U}_0[x] \left((s, \mathcal{P}), \Pi^{-1} E \right) \\ &= (\Pi_* \mathcal{U}_0)[x] \left((s, \mathcal{P}), E \right), \end{aligned} \quad (51)$$

so $\Pi_* \mathcal{U}_0$ is an update rule, which completes the proof. \square

In other words, an update pre-rule can be “pushed forward” to an update rule. If S and \mathcal{P}_N are finite, then an update rule can also be “pulled back” to an update pre-rule:

Proposition 2. If \mathcal{U} is an update rule and S and \mathcal{P}_N are finite, then there exists an update pre-rule, \mathcal{U}_0 , such that $\Pi_* \mathcal{U}_0 = \mathcal{U}$. That is, \mathcal{U} can be “pulled back” to \mathcal{U}_0 .

Proof. From \mathcal{U} , we define a map, \mathcal{U}_0 , as follows:

$$\mathcal{U}_0[x] \left((s, \mathcal{P}), (s', \mathcal{P}') \right) = \frac{1}{|\text{orb}_{\mathfrak{S}_N}(s', \mathcal{P}')|} \mathcal{U}[x] \left((s, \mathcal{P}), (s', \mathcal{P}') \bmod \mathfrak{S}_N \right). \quad (52)$$

For $\pi \in \mathfrak{S}_N$, we see from Eq. (40) that

$$\begin{aligned} \mathcal{U}_0[\pi x] \left((\pi s, \pi \mathcal{P}), (s', \mathcal{P}') \right) \\ = \frac{1}{|\text{orb}_{\mathfrak{S}_N}(s', \mathcal{P}')|} \mathcal{U}[\pi x] \left((\pi s, \pi \mathcal{P}), (s', \mathcal{P}') \bmod \mathfrak{S}_N \right) \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{|\text{orb}_{\mathfrak{S}_N}(s', \mathcal{P}')|} \mathcal{U}[x] \left((s, \mathcal{P}), (s', \mathcal{P}') \bmod \mathfrak{S}_N \right) \\
&= \mathcal{U}_0[x] \left((s, \mathcal{P}), (s', \mathcal{P}') \right).
\end{aligned} \tag{53}$$

Therefore, \mathcal{U}_0 satisfies Eq. (46) and defines an update pre-rule. Since

$$\begin{aligned}
&(\Pi_* \mathcal{U}_0)[x] \left((s, \mathcal{P}), (s', \mathcal{P}') \bmod \mathfrak{S}_N \right) \\
&= \mathcal{U}_0[x] \left((s, \mathcal{P}), \Pi^{-1} \left((s', \mathcal{P}') \bmod \mathfrak{S}_N \right) \right) \\
&= \sum_{(s'', \mathcal{P}'') \in \text{orb}_{\mathfrak{S}_N}(s', \mathcal{P}')} \mathcal{U}_0[x] \left((s, \mathcal{P}), (s'', \mathcal{P}'') \right) \\
&= \sum_{(s'', \mathcal{P}'') \in \text{orb}_{\mathfrak{S}_N}(s', \mathcal{P}')} \frac{1}{|\text{orb}_{\mathfrak{S}_N}(s'', \mathcal{P}'')|} \mathcal{U}[x] \left((s, \mathcal{P}), (s'', \mathcal{P}'') \bmod \mathfrak{S}_N \right) \\
&= \sum_{(s'', \mathcal{P}'') \in \text{orb}_{\mathfrak{S}_N}(s', \mathcal{P}')} \frac{1}{|\text{orb}_{\mathfrak{S}_N}(s', \mathcal{P}')|} \mathcal{U}[x] \left((s, \mathcal{P}), (s', \mathcal{P}') \bmod \mathfrak{S}_N \right) \\
&= \mathcal{U}[x] \left((s, \mathcal{P}), (s', \mathcal{P}') \bmod \mathfrak{S}_N \right),
\end{aligned} \tag{54}$$

it follows that $\Pi_* \mathcal{U}_0 = \mathcal{U}$, which completes the proof. \square

Remark 2. The proof of Proposition 2 requires that

$$E_0 \in \mathcal{F}(S^N \times \mathcal{P}_N) \implies \Pi E_0 \in \mathcal{F}(\mathcal{S}). \tag{55}$$

In the case that $S^N \times \mathcal{P}_N$ is finite, the singletons generate the canonical (i.e. discrete) σ -algebra on $S^N \times \mathcal{P}_N$, and the image $\Pi(s', \mathcal{P}') = (s', \mathcal{P}') \bmod \mathfrak{S}_N$ is measurable for each $(s', \mathcal{P}') \in S^N \times \mathcal{P}_N$. In general, the image of a measurable set need not be measurable. However, the purpose of introducing an update pre-rule is to provide an alternative way to obtain an update rule. An update rule defines a Markov chain on the *true* space of the evolutionary process, \mathcal{S} ; pulling this chain back to $S^N \times \mathcal{P}_N$ is not necessary.

An update pre-rule defines a Markov chain on $S^N \times \mathcal{P}_N$ whose kernel, κ_0 , satisfies

$$\kappa_0 \left((s, \mathcal{P}), E_0 \right) := \mathcal{U}_0 \left[F \left(u(s, \mathcal{P}) \right) \right] \left((s, \mathcal{P}), E_0 \right) \tag{56}$$

for each $(s, \mathcal{P}) \in S^N \times \mathcal{P}_N$ and $E_0 \in \mathcal{F}(S^N \times \mathcal{P}_N)$. Denote by $\Pi_* \kappa_0$ the kernel of the Markov chain defined by $\Pi_* \mathcal{U}_0$. The stationary distribution(s) of κ_0 can be “pushed forward” to stationary distribution(s) of $\Pi_* \kappa_0$ via the pushforward map of Eq. (49):

Proposition 3. If μ is a stationary distribution of the Markov chain defined by κ_0 , then $\Pi_* \mu$ is a stationary distribution of the Markov chain defined by $\Pi_* \kappa_0$.

Proof. Suppose that μ is a stationary distribution of κ_0 , i.e.

$$\mu(E_0) = \int_{\mathfrak{s} \in S^N \times \mathcal{P}_N} \kappa_0(\mathfrak{s}, E_0) d\mu(\mathfrak{s}) \tag{57}$$

for each $E_0 \in \mathcal{F}(S^N \times \mathcal{P}_N)$. For each $E \in \mathcal{F}(\mathcal{S})$, it follows that

$$\begin{aligned}
&\int_{\mathfrak{s} \bmod \mathfrak{S}_N \in \mathcal{S}} (\Pi_* \kappa_0)(\mathfrak{s} \bmod \mathfrak{S}_N, E) d(\Pi_* \mu)(\mathfrak{s} \bmod \mathfrak{S}_N) \\
&= \int_{\mathfrak{s} \in S^N \times \mathcal{P}_N} (\Pi_* \kappa_0)(\mathfrak{s} \bmod \mathfrak{S}_N, E) d\mu(\mathfrak{s}) \\
&= \int_{\mathfrak{s} \in S^N \times \mathcal{P}_N} \kappa_0(\mathfrak{s}, \Pi^{-1} E) d\mu(\mathfrak{s}) \\
&= \mu(\Pi^{-1} E) \\
&= (\Pi_* \mu)(E)
\end{aligned} \tag{58}$$

by the change of variables formula and Eq. (57), which completes the proof. \square

Thus, the passage from an update pre-rule to an update rule via Π_* is compatible with the steady states of the chains defined by κ_0 and $\Pi_*\kappa_0$, respectively.

3.3.2. Examples. We now give several classical examples of update pre-rules and rules:

Example 10 (Death-birth process). Suppose that S is finite and that \mathcal{P}_N is the finite subset of \mathcal{P}_N^G consisting of the undirected, unweighted graphs on N vertices (with no other restrictions—this set contains regular graphs, scale-free networks, etc.). In each step of a death-birth process, a player is selected uniformly at random from the population for death. The neighbors (determined by a graph, $\Gamma \in \mathcal{P}_N$) then compete to fill the vacancy: a neighbor—say, player j —is chosen for reproduction with probability proportional to relative fitness, x_j . The offspring of this player inherits the strategy of the parent and fills the vacancy left by the deceased player. The population state (i.e. the graph) is left unchanged by this process. We define an update pre-rule for this process by giving transition probabilities from $S^N \times \mathcal{P}_N$ to itself when the $N - 1$ surviving players in each round retain their labels. That is, if (s, Γ) is the state of the process and player i is chosen for death, Γ remains the same and only the i th coordinate of s is updated in order to obtain a new state, (s', Γ) . Thus, for two states, $(s, \Gamma), (s', \Gamma') \in S^N \times \mathcal{P}_N$, it must be the case that $\Gamma = \Gamma'$ for there to be a nonzero probability of transitioning from (s, Γ) to (s', Γ') . The probability of choosing player i for death is $1/N$, and, if this player is chosen for death, a transition is possible only if $s_j = s'_j$ for each $j \neq i$. The probability that player i is replaced by the offspring of a player using s_i' is

$$\sum_{j \neq i} \delta_{s_j, s'_i} \left(\frac{\Gamma_{ji} x_j}{\sum_{j \neq i} \Gamma_{ji} x_j} \right), \quad (59)$$

where δ_{s_j, s'_i} is 1 if $s_j = s'_i$ and 0 otherwise. Thus, for $x \in \mathbb{R}^N$, the transition probability from (s, Γ) to (s', Γ') is given by the update pre-rule, \mathcal{U}_0 , defined by

$$\mathcal{U}_0[x] \left((s, \Gamma), (s', \Gamma') \right) := \delta_{\Gamma, \Gamma'} \sum_{i=1}^N \left(\prod_{j \neq i} \delta_{s_j, s'_j} \right) \left(\frac{1}{N} \right) \left(\frac{\sum_{j \neq i} \delta_{s_j, s'_i} \Gamma_{ji} x_j}{\sum_{j \neq i} \Gamma_{ji} x_j} \right). \quad (60)$$

\mathcal{U}_0 is indeed an update pre-rule since, for each $\pi \in \mathfrak{S}_N$,

$$\begin{aligned} \mathcal{U}_0[\pi x] \left((\pi s, \pi \Gamma), (\pi s', \pi \Gamma') \right) &= \delta_{\pi \Gamma, \pi \Gamma'} \sum_{i=1}^N \left(\prod_{j \neq i} \delta_{s_{\pi(j)}, s'_{\pi(j)}} \right) \left(\frac{1}{N} \right) \left(\frac{\sum_{j \neq i} \delta_{s_{\pi(j)}, s'_{\pi(i)}} \Gamma_{\pi(j)\pi(i)} x_{\pi(j)}}{\sum_{j \neq i} \Gamma_{\pi(j)\pi(i)} x_{\pi(j)}} \right) \\ &= \delta_{\Gamma, \Gamma'} \sum_{i=1}^N \left(\prod_{j \neq \pi(i)} \delta_{s_j, s'_j} \right) \left(\frac{1}{N} \right) \left(\frac{\sum_{j \neq \pi(i)} \delta_{s_j, s'_{\pi(i)}} \Gamma_{j\pi(i)} x_j}{\sum_{j \neq \pi(i)} \Gamma_{j\pi(i)} x_j} \right) \\ &= \delta_{\Gamma, \Gamma'} \sum_{i=1}^N \left(\prod_{j \neq i} \delta_{s_j, s'_j} \right) \left(\frac{1}{N} \right) \left(\frac{\sum_{j \neq i} \delta_{s_j, s'_i} \Gamma_{ji} x_j}{\sum_{j \neq i} \Gamma_{ji} x_j} \right) \\ &= \mathcal{U}_0[x] \left((s, \Gamma), (s', \Gamma') \right). \end{aligned} \quad (61)$$

This example verifies in detail the symmetry condition of an update pre-rule, Eq. (46). Calculations for other processes are similar.

Example 11 (Wright-Fisher process). In contrast to the Moran process of Example 1 and the death-birth process of Example 10, one could also consider a process in which the entire population is updated synchronously. For example, in the *Wright-Fisher process* (Ewens, 2004; Imhof and Nowak, 2006), the population is updated as follows: A player—say, player i —is first selected for reproduction with probability proportional to fitness. The offspring of this player inherits the strategy of the parent with probability $1 - \varepsilon_i$ and takes on a novel strategy uniformly at random with probability ε_i . The mutation rate of the offspring is inherited from the parent (so that the offspring's offspring will also mutate with probability ε_i). This process is then repeated until there are N new offspring, and these offspring constitute the new population. Thus, one update step of the Wright-Fisher process involves updating the entire population.

Let S be finite and let \mathbf{m} be some finite subset of $[0, 1]$. The population state space for this version of the Wright-Fisher process is $\mathcal{P}_N := \mathbf{m}^N$. That is, a population state is an N -tuple of strategy-mutation rates, ε , with ε_i the mutation rate for player i . We define an update pre-rule, \mathcal{U}_0 , as follows: for $x \in \mathbb{R}^N$ and $(s, \varepsilon), (s', \varepsilon') \in S^N \times \mathcal{P}_N$, let

$$\mathcal{U}_0[x] \left((s, \varepsilon), (s', \varepsilon') \right) = \prod_{i=1}^N \sum_{j=1}^N \delta_{\varepsilon'_i, \varepsilon_j} \left(\frac{x_j}{x_1 + \dots + x_N} \right) \left[\delta_{s'_i, s_j} (1 - \varepsilon_j) + \varepsilon_j \left(\frac{1}{n} \right) \right]. \quad (62)$$

For each $\pi, \tau \in \mathfrak{S}_N$, $x \in \mathbb{R}^N$, and $(s, \varepsilon), (s', \varepsilon') \in S^N \times \mathcal{P}_N$, it is readily verified that

$$\mathcal{U}_0[\pi x] \left((\pi s, \pi \varepsilon), (\pi s', \pi \varepsilon') \right) = \mathcal{U}_0[x] \left((s, \varepsilon), (s', \varepsilon') \right). \quad (63)$$

Therefore, the resulting update rule, $\mathcal{U} := \Pi_* \mathcal{U}_0$, satisfies

$$\begin{aligned} \mathcal{U}[x] \left((s, \varepsilon) \bmod \mathfrak{S}_N, (s', \varepsilon') \bmod \mathfrak{S}_N \right) \\ = |\text{orb}_{\mathfrak{S}_N}(s', \varepsilon')| \prod_{i=1}^N \sum_{j=1}^N \delta_{\varepsilon'_i, \varepsilon_j} \left(\frac{x_j}{x_1 + \dots + x_N} \right) \left[\delta_{s'_i, s_j} (1 - \varepsilon_j) + \varepsilon_j \left(\frac{1}{n} \right) \right]. \end{aligned} \quad (64)$$

Consider the simple case $\varepsilon = \varepsilon' = \mathbf{0}$ (meaning there are no strategy mutations). If k'_r denotes the frequency of strategy r in state s' for $r = 1, \dots, n$, then $|\text{Stab}_{\mathfrak{S}_N}(s', \mathbf{0})| = k'_1! \dots k'_n!$, which means that

$$\begin{aligned} \mathcal{U}[x] \left((s, \mathbf{0}) \bmod \mathfrak{S}_N, (s', \mathbf{0}) \bmod \mathfrak{S}_N \right) \\ = |\text{orb}_{\mathfrak{S}_N}(s', \mathbf{0})| \prod_{i=1}^N \sum_{j=1}^N \left(\frac{x_j}{x_1 + \dots + x_N} \right) \delta_{s'_i, s_j} \\ = \frac{|\mathfrak{S}_N|}{|\text{Stab}_{\mathfrak{S}_N}(s', \mathbf{0})|} \prod_{i=1}^N \sum_{j=1}^N \left(\frac{x_j}{x_1 + \dots + x_N} \right) \delta_{s'_i, s_j} \\ = \binom{N}{k'_1, \dots, k'_n} \prod_{i=1}^N \sum_{j=1}^N \left(\frac{x_j}{x_1 + \dots + x_N} \right) \delta_{s'_i, s_j}, \end{aligned} \quad (65)$$

where the third line was obtained using the orbit-stabilizer theorem (see Knapp, 2006). Eq. (65) is just the classical formula for the transition probabilities of the Wright-Fisher process based on multinomial sampling (Kingman, 1980; Durrett, 2002; Der et al., 2011).

Example 12 (Pairwise comparison process). In each of our examples so far, both S and \mathcal{P}_N have been finite. Since our theory allows for these sets to be measurable, we now give an example of an evolutionary process whose strategy space is continuous. Let $S = [0, K]$ for some $K > 0$. This interval might be the strategy space for a public goods game, for instance, with K the maximum amount any one player may contribute to the public good. As in Examples 1 and 11, let $\mathcal{P}_N := \mathbf{m}^N$ for some finite subset, \mathbf{m} , of $[0, 1]$; an element of \mathcal{P}_N is just a profile of mutation rates, ε .

In each step of a pairwise comparison process, a player—say, player i —is selected uniformly at random from the population to evaluate his or her strategy. Another player—say, player j —is then chosen uniformly at random from the rest of the population as a model player. With probability $1 - \varepsilon_i$, the focal player takes into account the model player and probabilistically updates his or her strategy as follows: if x_i (resp. x_j) is the fitness of the focal (resp. model) player, and if $\beta \geq 0$ is the selection intensity, then the focal player imitates the model player with probability

$$\frac{1}{1 + e^{-\beta(x_j - x_i)}} \quad (66)$$

and retains his or her current strategy with probability

$$\frac{1}{1 + e^{-\beta(x_i - x_j)}} \quad (67)$$

(Szabó and Tóke, 1998). On the other hand, with probability ε_i the focal player ignores the model player completely. In this case, the focal player “explores” and adopts a new strategy from the interval $[0, K]$

probabilistically according to a truncated Gaussian distribution centered at s_i (the current strategy of the focal player). For some specified variance, σ^2 , this truncated Gaussian distribution has for a density function

$$\phi_{s_i}(x) := \left(\int_0^K \exp\left(-\frac{(y-s_i)^2}{2\sigma^2}\right) dy \right)^{-1} \exp\left(-\frac{(x-s_i)^2}{2\sigma^2}\right). \quad (68)$$

The parameter σ may be interpreted as a measure of how venturesome a player is, with cautious exploration corresponding to small σ and risky exploration corresponding to large σ . The density function, ϕ_{s_i} , defines a probability measure,

$$\begin{aligned} \Phi_{s_i} : \mathcal{F}(S) &\longrightarrow [0, 1] \\ : E &\longmapsto \int_E \phi_{s_i}(x) dx. \end{aligned} \quad (69)$$

If player i ignores the model player, then he or she adopts a strategy from $E \in \mathcal{F}(S)$ with probability $\Phi_{s_i}(E)$. Thus, a player who explores is more likely to adopt a strategy close to his or her current strategy than one farther away. For $\beta \geq 0$, let

$$g_\beta(x) := \frac{1}{1 + e^{-\beta x}} \quad (70)$$

be the logistic function. (In terms of this function, the probability that a focal player with fitness x_i imitates a model player with fitness x_j is $g_\beta(x_j - x_i)$.) We assemble these components into an update pre-rule, \mathcal{U}_0 , as follows: For $x \in \mathbb{R}^N$, $(s, \varepsilon) \in S^N \times \mathcal{P}_N$, and a measurable rectangle, $E_1 \times \cdots \times E_N \times E' \in \mathcal{F}(S)^N \times \mathcal{F}(\mathcal{P}_N)$, let

$$\begin{aligned} \mathcal{U}_0[x] &\left((s, \varepsilon), E_1 \times \cdots \times E_N \times E' \right) \\ &:= \delta_\varepsilon(E') \sum_{i=1}^N \left(\frac{1}{N} \right) \left(\prod_{j \neq i} \delta_{s_j}(E_j) \right) \sum_{j \neq i} \left(\frac{1}{N-1} \right) \left\{ \varepsilon_i \Phi_{s_i}(E_i) \right. \\ &\quad \left. + (1 - \varepsilon_i) \left[\delta_{s_j}(E_i) g_\beta(x_j - x_i) + \delta_{s_i}(E_i) g_\beta(x_i - x_j) \right] \right\}, \end{aligned} \quad (71)$$

and extend this definition additively to disjoint unions of measurable rectangles. For each $x \in \mathbb{R}^N$ and $(s, \varepsilon) \in S^N \times \mathcal{P}_N$, one can verify that $\mathcal{U}_0[x] \left((s, \varepsilon), - \right)$ extends to a measure on $S^N \times \mathcal{P}_N$ by the Hahn-Kolmogorov theorem, which we also denote by $\mathcal{U}_0[x] \left((s, \varepsilon), - \right)$. It is readily verified that \mathcal{U}_0 is an update pre-rule, so \mathcal{U}_0 extends to an update rule, $\Pi_* \mathcal{U}_0$, by Proposition 1. This example illustrates how the strategy mutations might themselves depend on the strategies (as opposed to simply being uniform random variables on S as they were in the previous examples).

In Examples 1, 11, and 12, we considered processes with heterogeneous mutation rates (meaning ε_i depends on i). In Examples 1 and 11, there is a nonzero probability of transitioning from a state with heterogeneous mutation rates to a state with homogeneous mutation rates. For instance, in the Wright-Fisher process of Example 11, if (s, ε) is a state such that $\varepsilon_\ell \neq \varepsilon_{\ell'}$ for some ℓ and ℓ' , and if (s', ε') is a state satisfying $s'_1 = s'_2 = \cdots = s'_N = s_\ell$ and $\varepsilon'_1 = \varepsilon'_2 = \cdots = \varepsilon'_N = \varepsilon_\ell$, then there is a nonzero probability of transitioning from (s, ε) to (s', ε') provided $x_\ell > 0$. Thus, the population state, which is simply of profile of mutation rates, can change from generation to generation. In contrast, the population state of Example 12 *cannot* change from generation to generation since strategies are imitated and mutation rates are not inherited. In other words, much of the biological meaning behind the quantities appearing in the population state are encoded in the dynamics of the process via the update rule.

In a more formal setting, let κ be the transition kernel obtained from an update rule via Eq. (43). From the projection $\Pi_2 : S^N \times \mathcal{P}_N \rightarrow \mathcal{P}_N$, we obtain a map

$$\begin{aligned} \tilde{\Pi}_2 : (S^N \times \mathcal{P}_N) / \mathfrak{S}_N &\longrightarrow \mathcal{P}_N / \mathfrak{S}_N \\ : (s, \mathcal{P}) \bmod \mathfrak{S}_N &\longmapsto \mathcal{P} \bmod \mathfrak{S}_N. \end{aligned} \quad (72)$$

$\tilde{\Pi}_2$ gives us a pushforward map, $\left(\tilde{\Pi}_2\right)_* : \Delta(\mathcal{S}) \rightarrow \Delta(\mathcal{P}_N/\mathfrak{S}_N)$, which we use to formalize the intuition behind “static” and “dynamic” population states:

Definition 6 (Static and dynamic population states). A population state, \mathcal{P} , in a population state space, \mathcal{P}_N , is *static* relative to κ if, for each $s \in S^N$,

$$\left(\tilde{\Pi}_2\right)_* \kappa\left((s, \mathcal{P}) \bmod \mathfrak{S}_N, -\right) = \delta_{\mathcal{P} \bmod \mathfrak{S}_N}, \quad (73)$$

where $\delta_{\mathcal{P} \bmod \mathfrak{S}_N}$ denotes the Dirac measure on $\Delta(\mathcal{P}_N/\mathfrak{S}_N)$ centered at $\mathcal{P} \bmod \mathfrak{S}_N$. Otherwise, if κ is not static relative to κ , we say that \mathcal{P} is *dynamic* relative to κ .

In Examples 10 and 12, every population state is static. In Examples 1 and 11, only the population states (i.e. mutation profiles) with $\varepsilon_1 = \varepsilon_2 = \dots = \varepsilon_N$ are static.

4. STOCHASTIC SELECTION PROCESSES WITH VARIABLE POPULATION SIZE

Suppose now that the population size is dynamic, and let $\mathcal{N} \subseteq \{0, 1, 2, \dots\}$ be the set of admissible population sizes. As in §3, let S be the strategy space for each player. Instead of having a single population state space, we now require the existence of a population state space, \mathcal{P}_N , for *each* $N \in \mathcal{N}$. The state space for such a process is

$$\mathcal{S} := \bigsqcup_{\ell \in \mathcal{N}} (S^\ell \times \mathcal{P}_\ell) / \mathfrak{S}_\ell, \quad (74)$$

where $\bigsqcup_{\ell \in \mathcal{N}} (S^\ell \times \mathcal{P}_\ell) / \mathfrak{S}_\ell$ denotes the disjoint union of the spaces $(S^\ell \times \mathcal{P}_\ell) / \mathfrak{S}_\ell$. Instead of a single aggregate payoff function and update rule, we now require that there be an aggregate payoff function, $u^N : S^N \times \mathcal{P}_N \rightarrow \mathbb{R}^N$, and an update rule, $\mathcal{U}^N : \mathbb{R}^N \rightarrow K(S^N \times \mathcal{P}_N, \mathcal{S})$, for *each* admissible population size, $N \in \mathcal{N}$. If the population currently has size N , then u^N determines the payoffs to the players in the interaction step, and \mathcal{U}^N updates the population (possibly to one of a different size). Of course, for each $\pi \in \mathfrak{S}_N$, $x \in \mathbb{R}^N$, $s \in S^N$, $\mathcal{P} \in \mathcal{P}_N$, and $E \in \mathcal{F}(\mathcal{S})$, these functions must satisfy

$$u^N(\pi s, \pi \mathcal{P}) = \pi u^N(s, \mathcal{P}); \quad (75a)$$

$$\mathcal{U}^N[\pi x] \left((\pi s, \pi \mathcal{P}), E \right) = \mathcal{U}^N[x] \left((s, \mathcal{P}), E \right) \quad (75b)$$

just as they did in Definitions 3 and 4, respectively.

Finally, we have the definition of a stochastic selection process in its full generality:

Definition 7 (Stochastic selection process). A *stochastic selection process* consists of the following components:

- (1) a set of admissible population sizes, $\mathcal{N} \subseteq \{0, 1, 2, \dots\}$;
- (2) a measurable strategy space, S ;
- (3) for each $N \in \mathcal{N}$, a population state space, \mathcal{P}_N ;
- (4) for each $N \in \mathcal{N}$, an aggregate payoff function, $u^N : S^N \times \mathcal{P}_N \rightarrow \mathbb{R}^N$;
- (5) a payoff-to-fitness function, $f : \mathbb{R} \rightarrow \mathbb{R}$;
- (6) for each $N \in \mathcal{N}$, an update rule, $\mathcal{U}^N : \mathbb{R}^N \rightarrow K(S^N \times \mathcal{P}_N, \mathcal{S})$, where

$$\mathcal{S} := \bigsqcup_{\ell \in \mathcal{N}} (S^\ell \times \mathcal{P}_\ell) / \mathfrak{S}_\ell. \quad (76)$$

The components of a stochastic selection process produce a Markov chain on \mathcal{S} whose kernel, κ , is defined as follows: for $N \in \mathcal{N}$, $(s, \mathcal{P}) \in S^N \times \mathcal{P}_N$, and $E \in \mathcal{S}$,

$$\kappa\left((s, \mathcal{P}) \bmod \mathfrak{S}_N, E\right) = \mathcal{U}^N\left[F\left(u^N(s, \mathcal{P})\right)\right]\left((s, \mathcal{P}), E\right). \quad (77)$$

It is readily verified that κ is well defined (see Eq. (44)).

Remark 3. The payoff-to-fitness function, f , in requirement (5) of a stochastic selection process, is not strictly necessary. It could instead be absorbed into either the payoff function (which would then be a *fitness function*) or the update rule (which would then be a family of transition kernels parametrized by *payoff profiles* rather than by fitness profiles). We include this function as a part of a stochastic selection process

for three reasons: (1) having an aggregate payoff function instead of an aggregate fitness function allows for a more straightforward comparison to the theory of stochastic games; (2) having an update rule be a family of transition kernels parametrized by fitness simplifies its presentation (see §3.3.2); and (3) payoff-to-fitness functions are often explicitly mentioned in models of evolutionary games in the literature. Tuning the selection strength of a process, for instance, amounts to modifying the payoff-to-fitness function, so including this function in a stochastic selection process allows one to more explicitly separate the various components of a selection process.

Remark 4. The notion of *update pre-rule* also makes sense for populations of variable size, although one must define the symmetry condition, Eq. (46), with greater care. If \mathcal{N} is the set of admissible population sizes, then we require—for each $N \in \mathcal{N}$ —a map

$$\mathcal{U}_0^N : \mathbb{R}^N \longrightarrow K \left(S^N \times \mathcal{P}_N, \bigsqcup_{\ell \in \mathcal{N}} S^\ell \times \mathcal{P}_\ell \right). \quad (78)$$

For each $N \in \mathcal{N}$, there is an action of \mathfrak{S}_N on $\bigsqcup_{\ell \in \mathcal{N}} S^\ell \times \mathcal{P}_\ell$ defined by

$$\pi(s, \mathcal{P}) = \begin{cases} (\pi s, \pi \mathcal{P}) & (s, \mathcal{P}) \in S^N \times \mathcal{P}_N; \\ (s, \mathcal{P}) & (s, \mathcal{P}) \notin S^N \times \mathcal{P}_N. \end{cases} \quad (79)$$

From the set of groups $\{\mathfrak{S}_N\}_{N \in \mathcal{N}}$, one can construct the *free product*,

$$\mathfrak{S}_{\mathcal{N}} := \bigast_{N \in \mathcal{N}} \mathfrak{S}_N, \quad (80)$$

which is just the analogue of disjoint union in the category of groups (see Knapp, 2006). Collectively, the actions of $S^N \times \mathcal{P}_N$ on $\bigsqcup_{\ell \in \mathcal{N}} S^\ell \times \mathcal{P}_\ell$ defined by Eq. (79) (over all $N \in \mathcal{N}$) result in a (measurable) action of $\mathfrak{S}_{\mathcal{N}}$ on $\bigsqcup_{\ell \in \mathcal{N}} S^\ell \times \mathcal{P}_\ell$. For $\{\mathcal{U}_0^N\}_{N \in \mathcal{N}}$ to define a collection of update pre-rules, we require that for each $N \in \mathcal{N}$, $\pi \in \mathfrak{S}_N$, $x \in \mathbb{R}^N$, $(s, \mathcal{P}) \in S^N \times \mathcal{P}_N$ and $E_0 \in \mathcal{F}(\bigsqcup_{\ell \in \mathcal{N}} S^\ell \times \mathcal{P}_\ell)$, there exists $\tau \in \mathfrak{S}_{\mathcal{N}}$ such that

$$\mathcal{U}_0[\pi x] \left((\pi s, \pi \mathcal{P}), E_0 \right) = \mathcal{U}_0[x] \left((s, \mathcal{P}), \tau E_0 \right). \quad (81)$$

The reason we require τ to be in the free product, $\mathfrak{S}_{\mathcal{N}}$, instead of just in \mathfrak{S}_N for some $N \in \mathcal{N}$, is that it need not hold that $E_0 \in \mathcal{F}(S^N \times \mathcal{P}_N)$ for some $N \in \mathcal{N}$. E_0 could be some complicated measurable set consisting of elements of $S^N \times \mathcal{P}_N$ for *several* $N \in \mathcal{N}$, so we need some way of relabeling elements $S^N \times \mathcal{P}_N$ for several values of $N \in \mathcal{N}$ simultaneously. Extending the action of \mathfrak{S}_N on $S^N \times \mathcal{P}_N$ to $\bigsqcup_{\ell \in \mathcal{N}} S^\ell \times \mathcal{P}_\ell$ via (79), in order to form the free product, $\mathfrak{S}_{\mathcal{N}}$, via Eq. (80), accomplishes this task.

Analogues of Propositions 1, 2, and 3 also hold in this context, but we do not go through the details here; the proofs are essentially the same as they were in §3.3.1.

5. DISCUSSION

We use the term “selection process” instead of “evolutionary game” in order to emphasize that the update step is based on the principles of natural selection. Several types of adaptive processes appearing in the economics literature have been referred to as evolutionary games. *Best-response dynamics* of Ellison (1993) is a procedure in which, at each round, the players update their strategies based on the best responses to their opponents in the previous round. This process is known to converge to a Nash equilibrium of the game. Hart and Mas-Colell (2000) define a similar process called *regret matching* that leads to a correlated equilibrium of the game. These processes can be phrased as stochastic games (along with appropriate strategies), but they are *not* stochastic selection processes, as we now illustrate with best-response dynamics:

Suppose $N = 2$. Let $S = \{A, B\}$ and let $u : S^2 \rightarrow \mathbb{R}^2$ be the payoff function for a game between two players. If best-response dynamics in this population defines a stochastic selection process, then there exists a population state space, \mathcal{P}_2 , and an update rule, \mathcal{U} , such that the transition kernel of the resulting Markov chain, κ_u , satisfies

$$\kappa_u \left((s, \mathcal{P}), E \right) = \mathcal{U}[u(s)] \left((s, \mathcal{P}), E \right) \quad (82)$$

for each $(s, \mathcal{P}) \in S^2 \times \mathcal{P}_2$ and $E \in \mathcal{F}(S)$. In other words, κ_u depends on u insofar as \mathcal{U} depends on \mathbb{R}^2 . Consider the two payoff functions, $u, v : S^2 \rightarrow \mathbb{R}^2$, defined by

$$\begin{pmatrix} u(A, A) & u(A, B) \\ u(B, A) & u(B, B) \end{pmatrix} = \begin{pmatrix} 2 & 2 \\ 1 & 1 \end{pmatrix}; \quad (83a)$$

$$\begin{pmatrix} v(A, A) & v(A, B) \\ v(B, A) & v(B, B) \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix}. \quad (83b)$$

Since $u(B, B) = v(B, B) = 1$, it follows that for $s = (B, B)$ and any $\mathcal{P} \in \mathcal{P}_2$,

$$\begin{aligned} \kappa_u \left(\left((B, B), \mathcal{P} \right), E \right) &= \mathcal{U}[u(B, B)] \left(\left((B, B), \mathcal{P} \right), E \right) \\ &= \mathcal{U}[v(B, B)] \left(\left((B, B), \mathcal{P} \right), E \right) \\ &= \kappa_v \left(\left((B, B), \mathcal{P} \right), E \right). \end{aligned} \quad (84)$$

Thus, the strategy profile (B, B) must be updated by best-response dynamics in the same way for both functions, u and v . However, best-response dynamics actually results in different updates of (B, B) for these two games: for u , the profile (B, B) is updated to (A, A) ; for v , the profile (B, B) is updated to (B, B) (it is already a Nash equilibrium). The key observation is that, while the Markov chain defined by a stochastic selection process depends on the aggregate payoff function, u , the update rule is *independent* of u . In contrast, the update step in best-response dynamics clearly depends on u . In a stochastic selection process, the only role of the aggregate payoff function is to determine the fitness profile (which is then passed to the update rule).

An update rule in the classical sense, vaguely speaking, generally consists of information about births and deaths (or imitation). Choosing to represent the strategies of the players in the population as an N -tuple, $s \in S^N$, is just a mathematical convenience; the update rule is independent of how the players are labeled. We defined the notion of update pre-rule (Definition 5) in order to relate stochastic selection processes to the way in which evolutionary processes are frequently modeled—as Markov chains on S^N (or, more generally, on $S^N \times \mathcal{P}_N$). In many cases, an update pre-rule is simpler to write down explicitly than an update rule since one can choose a convenient enumeration of the players in each time step. We showed that an update pre-rule can always be “pushed forward” to an update rule, so it is sufficient to give an update pre-rule in place of an update rule in the definition of stochastic selection process. However, an update *rule* is the true mathematical formalization of an evolutionary update in this context since it is independent of the labeling of the players.

Stochastic selection processes encompass existing models of selection such as the *evolutionary game Markov chain* of Wage (2010) and the *evolutionary Markov chain* of Allen and Tarnita (2012). Wage (2010) considers a Markov chain arising from probability distributions over a collection of *inheritance rules*. An inheritance rule is a map, $I : \{1, \dots, N\} \rightarrow \{1, \dots, N, m\}$, that designates the source of a player’s strategy: if $I(i) = j \neq m$, then player i inherits his or her strategy from player j ; if $I(i) = m$, then player i ’s strategy is the result of a random mutation (assumed to be uniform over the strategy set). Similarly, Allen and Tarnita (2012) model evolution in populations with fixed size and structure using *replacement events*. A replacement event is a pair, (R, α) , consisting of a collection, $R \subseteq \{1, \dots, N\}$, of players who are replaced and a rule, $\alpha : R \rightarrow \{1, \dots, N\}$, indicating the parent of each offspring involved in the replacement. These frameworks provide good models for many classical evolutionary processes, but they do not account for genetic processes with crossover, for instance. Moreover, one could imagine a cultural process in which a player updates his or her strategy based on some complicated synthesis of *many* strategies in the population. Our framework generalizes these models, taking into account arbitrary strategy spaces, payoff functions, population structures, and update rules.

McAvoy and Hauert (2015b) define the notion of a *homogeneous* evolutionary game, which, informally, means that any two states consisting of a single A -mutant in a monomorphic B -population are equivalent. More specifically, suppose that a population state consists of a graph and a profile of mutation rates; that is, $\mathcal{P}_N = \mathcal{P}_N^G \times \mathbf{m}^N$ for some finite subset, \mathbf{m} , of $[0, 1]$. If $s, s' \in S^N$, and if $\pi \in \mathfrak{S}_N$ satisfies $\pi\Gamma = \Gamma$ and $\pi\varepsilon = \varepsilon$ for some $\Gamma \in \mathcal{P}_N^G$ and $\varepsilon \in \mathbf{m}^N$, then McAvoy and Hauert (2015b) argue that the states $(s, (\Gamma, \varepsilon))$ and $(\pi s, (\Gamma, \varepsilon))$ are *evolutionarily equivalent* in the sense that π induces an automorphism of the Markov chain on $S^N \times \mathcal{P}_N$

that sends $(s, (\Gamma, \varepsilon))$ to $(\pi s, (\Gamma, \varepsilon))$. This result is a special case of an observation that is completely obvious in the context of stochastic selection processes: if $\pi \in \text{Aut}(\mathcal{P})$ for some \mathcal{P} in a population state space, \mathcal{P}_N , then the representatives (s, \mathcal{P}) and $(\pi s, \mathcal{P}) = (\pi s, \pi \mathcal{P})$ define *exactly the same state* in \mathcal{S} . Therefore, working on the true state space of an evolutionary process, \mathcal{S} , helps to elucidate structural symmetries that are not as clear when working with a Markov chain on $S^N \times \mathcal{P}_N$ that is defined by an update pre-rule.

In the definition of a population state space (Definition 1), we require a measurable space, \mathcal{P}_N , along with a measurable action of \mathfrak{S}_N on \mathcal{P}_N . Naturally, this setup raises the question of what types of \mathfrak{S}_N -actions one can put on \mathcal{P}_N while still retaining the desired properties of an evolutionary process. If one were to take an update rule, \mathcal{U} , and then arbitrarily change the action of \mathfrak{S}_N on \mathcal{P}_N , then \mathcal{U} need not remain an update rule under this new action. For example, let $\Gamma^0 \in \mathcal{P}_N^G$ be the Frucht graph, which is an undirected, unweighted, regular graph with $N = 12$ vertices and no nontrivial symmetries (Frucht, 1939). In other words, if $\Gamma_{\pi(i)\pi(j)}^0 = \Gamma_{ij}^0$ for each $i, j = 1, \dots, 12$, then $\pi = \text{id}$. Instead of the standard action of \mathfrak{S}_N on \mathcal{P}_N^G , one could instead declare that \mathfrak{S}_N acts trivially on \mathcal{P}_N^G ; in particular, $\pi \star \Gamma = \Gamma$ for each $\pi \in \mathfrak{S}_N$ and $\Gamma \in \mathcal{P}_N^G$. If \mathcal{U} is the update rule for the death-birth process (see Example 10), then it follows that the representatives (s, Γ^0) and $(\pi s, \pi \star \Gamma^0) = (\pi s, \Gamma^0)$ define the same point in the state space, \mathcal{S} . For the (frequency-dependent) Snowdrift Game, all 12 states consisting of a single cooperator in a population of defectors give rise to different fixation probabilities for cooperators (see McAvoy and Hauert, 2015b). Therefore, it cannot be the case that \mathcal{U} defines a Markov chain on \mathcal{S} via Eq. (43); in particular, \mathcal{U} is no longer a well-defined update rule under the new action, \star , of \mathfrak{S}_N on \mathcal{P}_N .

The dynamics of a stochastic selection process, which are obtained via its update rule, encode much of the biological meaning of the components that constitute the population state space. In both of Examples 11 and 12, the population state space was $\mathcal{P}_N := \mathbf{m}^N$ for some finite subset, \mathbf{m} , of $[0, 1]$. On the other hand, the interpretations of these mutation rates were completely different in these two processes: in Example 11, the interpretation applied to the offspring of reproducing players; in Example 12, the mutation was interpreted as “exploration” and applied to a player who was chosen to update his or her strategy. One could even consider different implementations of mutation rates in the same process: in a genetic process based on reproduction, a player’s strategy-mutation rate may be inherited from the parent (as in Examples 1 and 11), or, alternatively, it may be determined by a player’s spatial location (see McAvoy and Hauert, 2015b). These details are encoded entirely in the update rule.

Although the framework we present here is clearly aimed at evolutionary *games* used to describe natural selection, related processes that are not technically “games” may also constitute stochastic selection processes. Evolutionary algorithms, for example, form an important subclass of stochastic selection processes. These algorithms seek to apply the principles of natural selection to solve search and optimization problems (Bäck, 1996). Evolutionary algorithms typically do not have population state spaces, which, in our context, means that \mathcal{P}_N can be taken to be a singleton equipped with the trivial action of \mathfrak{S}_N . A popular type of evolutionary algorithm, known as a *genetic* algorithm, involves representing the elements of the search space, i.e. the *genomes* in S , as sequences of binary digits. Each genome is then assigned a fitness based on its viability as a solution to the problem at hand. (Unlike in biological populations, the fitness landscape, although complex, is inherently static and does not depend on the other members of the population.) The update step, which is commonly designed to mimic sexual reproduction in nature, involves a combination of selection, crossover, and mutation. A population of genomes is then repeatedly updated until a sufficiently fit genome appears. Despite the fact that biological reproduction generally involves either one (asexual) or two (sexual) parents, evolutionary algorithms have been simulated using *many* parents (Chambers, 1998). Other components of the update step in some algorithms, such as stochastic universal sampling (Baker, 1987), elitism (Baluja and Caruana, 1995), and tournament selection (Poli, 2005), are all readily incorporated into our model of stochastic selection processes.

In Example 12, we saw an evolutionary process with an uncountably infinite state space. A state space of this sort arises naturally in the public goods game, for instance, where there is a continuous range of investment levels. This example also illustrates the more complicated ways in which strategy mutations can be incorporated into an evolutionary process. If a player has a strategy, $x \in [0, K]$ for some $K > 0$, then it may be the case that this player is more likely to “explore” strategies close to x than he or she is to switch to strategies farther away. A truncated Gaussian random variable on $[0, K]$, whose variance is a measure of how venturesome a player is, captures this type of strategy exploration and is easily incorporated

into the update rule of an evolutionary process. This type of mutation has appeared in the context of adaptive dynamics (Doebeli et al., 2004) and, more recently, in a study of stochastic evolutionary branching (Wakano and Iwasa, 2012), but it has been largely ignored elsewhere in the literature on evolutionary game theory, where strategy mutations typically involve switching between two strategies or else are governed by a uniform random variable over the strategy space. Further studies of the dynamics of processes with these biologically-relevant mutations are certainly warranted.

Our framework makes no assumptions on the cardinality of S and \mathcal{P}_N ; all that is required is that these spaces be measurable (and that \mathcal{P}_N be equipped with an action of \mathfrak{S}_N). Markov chains on continuous state spaces have unique stationary distributions under certain circumstances (see Durrett, 2009), but, in the generality of this framework, it need not be the case that a stationary distribution is unique. Even if S is finite and there are nonzero strategy-mutation rates, the spatial structure of the population might be disconnected, resulting in multiple stationary distributions. Particular instances of stochastic selection processes may have the property that nonzero strategy-mutation rates imply that the Markov chain defined by the process is irreducible (Fudenberg and Imhof, 2006, 2008; Allen and Tarnita, 2012), but this phenomenon need not hold in general. Our goal here was not to study the dynamics of any particular subclass of stochastic selection processes, but rather to formalize *what these processes are*.

Our general theory of stochastic selection processes provides a mathematical foundation for a broad class of processes used to describe evolution by means of natural selection in finite populations. Stochastic selection processes also provide a mathematical framework for processes with *variable* population size, a topic that has received surprisingly little attention in the literature. Although many biological interactions have been modeled using classical games, the differences between stochastic games and stochastic selection processes illustrate a fundamental distinction between classical and evolutionary game theory. There is still a lot to discover about the dynamics of selection processes in finite populations (especially those with variable population size), and our hope is that this framework elucidates the roles of the components of processes based on natural selection and advances the effort to transform evolution into a mathematical theory.

ACKNOWLEDGMENTS

The author thanks Christoph Hauert for many helpful discussions and for carefully reading earlier versions of this manuscript. Financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) is gratefully acknowledged.

REFERENCES

- B. Allen and C. E. Tarnita. Measures of success in a class of evolutionary models with fixed population size and structure. *Journal of Mathematical Biology*, 68(1-2):109–143, Nov 2012. doi: 10.1007/s00285-012-0622-x.
- T. Antal, H. Ohtsuki, J. Wakeley, P. D. Taylor, and M. A. Nowak. Evolution of cooperation by phenotypic similarity. *Proceedings of the National Academy of Sciences*, 106(21):8597–8600, Apr 2009. doi: 10.1073/pnas.0902528106.
- R. J. Aumann. Correlated equilibrium as an expression of bayesian rationality. *Econometrica*, 55(1):1, Jan 1987. doi: 10.2307/1911154.
- T. Bäck. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, Oxford, UK, 1996. ISBN 0-19-509971-0.
- J. E. Baker. Reducing bias and inefficiency in the selection algorithm. In *Proceedings of the Second International Conference on Genetic Algorithms on Genetic Algorithms and Their Application*, pages 14–21, Hillsdale, NJ, USA, 1987. L. Erlbaum Associates Inc. ISBN 0-8058-0158-8.
- S. Baluja and R. Caruana. Removing the genetics from the standard genetic algorithm. In *Machine Learning Proceedings 1995*, pages 38–46. Elsevier, 1995. doi: 10.1016/b978-1-55860-377-6.50014-1.
- L. Chambers, editor. *Practical Handbook of Genetic Algorithms*. CRC Press, Dec 1998. doi: 10.1201/9781420050080.
- F. Débarre, C. Hauert, and M. Doebeli. Social evolution in structured populations. *Nature Communications*, 5, Mar 2014. doi: 10.1038/ncomms4409.

- R. Der, C. L. Epstein, and J. B. Plotkin. Generalized population models and the nature of genetic drift. *Theoretical Population Biology*, 80(2):80–99, Sep 2011. doi: 10.1016/j.tpb.2011.06.004.
- M. Doebeli, C. Hauert, and T. Killingback. The Evolutionary Origin of Cooperators and Defectors. *Science*, 306(5697):859–862, Oct 2004. doi: 10.1126/science.1101456.
- L. A. Dugatkin. *Game Theory and Animal Behavior*. Oxford University Press, 2000.
- R. Durrett. *Probability Models for DNA Sequence Evolution*. Springer New York, 2002. doi: 10.1007/978-1-4757-6285-3.
- R. Durrett. *Probability: Theory and Examples*. Cambridge University Press, 2009. doi: 10.1017/cbo9780511779398.
- G. Ellison. Learning, local interaction, and coordination. *Econometrica*, 61(5):1047, Sep 1993. doi: 10.2307/2951493.
- W. J. Ewens. *Mathematical Population Genetics*. Springer New York, 2004. doi: 10.1007/978-0-387-21822-9.
- R. Frucht. Herstellung von graphen mit vorgegebener abstrakter gruppe. *Compositio Mathematica*, 6:239–250, 1939.
- D. Fudenberg and L. A. Imhof. Imitation processes with small mutations. *Journal of Economic Theory*, 131(1):251–262, Nov 2006. doi: 10.1016/j.jet.2005.04.006.
- D. Fudenberg and L. A. Imhof. Monotone imitation dynamics in large populations. *Journal of Economic Theory*, 140(1):229–245, May 2008. doi: 10.1016/j.jet.2007.08.002.
- D. Fudenberg and J. Tirole. *Game Theory*. The MIT Press, 1991.
- S. Hart and A. Mas-Colell. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5):1127–1150, Sep 2000. doi: 10.1111/1468-0262.00153.
- C. Hauert and L. Imhof. Evolutionary games in deme structured, finite populations. *Journal of Theoretical Biology*, 299:106–112, Apr 2012. doi: 10.1016/j.jtbi.2011.06.010.
- C. Hauert and H. G. Schuster. Effects of increasing the number of players and memory size in the iterated prisoner’s dilemma: a numerical approach. *Proceedings of the Royal Society of London B: Biological Sciences*, 264(1381):513–519, 1997. doi: 10.1098/rspb.1997.0073.
- J. Hofbauer and K. Sigmund. *Evolutionary Games and Population Dynamics*. Cambridge University Press, 1998. doi: 10.1017/cbo9781139173179.
- L. A. Imhof and M. A. Nowak. Evolutionary game dynamics in a wright-fisher process. *Journal of Mathematical Biology*, 52(5):667–681, Feb 2006. doi: 10.1007/s00285-005-0369-8.
- J. F. C. Kingman. *Mathematics of Genetic Diversity*. Society for Industrial and Applied Mathematics, Jan 1980. doi: 10.1137/1.9781611970357.
- A. W. Knap. *Basic Algebra*. Birkhuser Boston, 2006. doi: 10.1007/978-0-8176-4529-8.
- E. Lieberman, C. Hauert, and M. A. Nowak. Evolutionary dynamics on graphs. *Nature*, 433(7023):312–316, Jan 2005. doi: 10.1038/nature03204.
- W. Maciejewski, F. Fu, and C. Hauert. Evolutionary game dynamics in populations with heterogenous structures. *PLoS Computational Biology*, 10(4):e1003567, Apr 2014. doi: 10.1371/journal.pcbi.1003567.
- J. Maynard Smith. *Evolution and the Theory of Games*. Cambridge University Press, 1982. doi: 10.1017/cbo9780511806292.
- A. McAvoy and C. Hauert. Structure coefficients and strategy selection in multiplayer games. *Journal of Mathematical Biology*, Apr 2015a. doi: 10.1007/s00285-015-0882-3.
- A. McAvoy and C. Hauert. Structural symmetry in evolutionary games. *Journal of The Royal Society Interface*, 12(111):20150420, Sep 2015b. doi: 10.1098/rsif.2015.0420.
- A. McAvoy and C. Hauert. Asymmetric evolutionary games. *PLOS Computational Biology*, 11(8):e1004349, Aug 2015c. doi: 10.1371/journal.pcbi.1004349.
- J. McMillan. *Game Theory in International Economics*. Routledge, Dec 2001. doi: 10.4324/9781315014562.
- J.-F. Mertens, S. Sorin, and S. Zamir. *Repeated Games*. Cambridge University Press, 2015. doi: 10.1017/cbo9781139343275.
- P. A. P. Moran. Random processes in genetics. *Mathematical Proceedings of the Cambridge Philosophical Society*, 54(01):60, Jan 1958. doi: 10.1017/s0305004100033193.
- A. Neyman. From markov chains to stochastic games. In *Stochastic Games and Applications*, pages 9–25. Springer Netherlands, 2003. doi: 10.1007/978-94-010-0189-2\$2.

- A. Neyman and S. Sorin, editors. *Stochastic Games and Applications*. Springer Netherlands, 2003. doi: 10.1007/978-94-010-0189-2.
- M. A. Nowak. *Evolutionary Dynamics: Exploring the Equations of Life*. Belknap Press, 2006.
- M. A. Nowak, A. Sasaki, C. Taylor, and D. Fudenberg. Emergence of cooperation and evolutionary stability in finite populations. *Nature*, 428(6983):646–650, Apr 2004. doi: 10.1038/nature02414.
- M. A. Nowak, C. E. Tarnita, and T. Antal. Evolutionary dynamics in structured populations. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 365(1537):19–30, Nov 2009. doi: 10.1098/rstb.2009.0215.
- H. Ohtsuki and M. A. Nowak. The replicator equation on graphs. *Journal of Theoretical Biology*, 243(1): 86–97, Nov 2006. doi: 10.1016/j.jtbi.2006.06.004.
- H. Ohtsuki, C. Hauert, E. Lieberman, and M. A. Nowak. A simple rule for the evolution of cooperation on graphs and social networks. *Nature*, 441(7092):502–505, May 2006. doi: 10.1038/nature04605.
- H. Ohtsuki, M. A. Nowak, and J. M. Pacheco. Breaking the symmetry between interaction and replacement in evolutionary dynamics on graphs. *Physical Review Letters*, 98(10), Mar 2007a. doi: 10.1103/physrevlett.98.108106.
- H. Ohtsuki, J. M. Pacheco, and M. A. Nowak. Evolutionary graph theory: Breaking the symmetry between interaction and replacement. *Journal of Theoretical Biology*, 246(4):681–694, Jun 2007b. doi: 10.1016/j.jtbi.2007.01.024.
- J. M. Pacheco, F. L. Pinheiro, and F. C. Santos. Population structure induces a symmetry breaking favoring the emergence of cooperation. *PLoS Computational Biology*, 5(12):e1000596, Dec 2009. doi: 10.1371/journal.pcbi.1000596.
- R. Poli. Tournament selection, iterated coupon-collection problem, and backward-chaining evolutionary algorithms. In *Foundations of Genetic Algorithms*, pages 132–155. Springer Science + Business Media, 2005. doi: 10.1007/11513575\$_8\$.
- W. H. Press and F. J. Dyson. Iterated prisoner’s dilemma contains strategies that dominate any evolutionary opponent. *Proceedings of the National Academy of Sciences*, 109(26):10409–10413, May 2012. doi: 10.1073/pnas.1206569109.
- M. L. Puterman, editor. *Markov Decision Processes*. John Wiley & Sons, Inc., Apr 1994. doi: 10.1002/9780470316887.
- D. G. Rand, M. A. Nowak, J. H. Fowler, and N. A. Christakis. Static network structure can stabilize human cooperation. *Proceedings of the National Academy of Sciences*, 111(48):17093–17098, Nov 2014. doi: 10.1073/pnas.1400406111.
- L. S. Shapley. Stochastic games. *Proceedings of the National Academy of Sciences*, 39(10):1095–1100, Oct 1953. doi: 10.1073/pnas.39.10.1095.
- G. Szabó and G. Fáth. Evolutionary games on graphs. *Physics Reports*, 446(4-6):97–216, Jul 2007. doi: 10.1016/j.physrep.2007.04.004.
- G. Szabó and C. Tóke. Evolutionary prisoner’s dilemma game on a square lattice. *Physical Review E*, 58(1):69–73, Jul 1998. doi: 10.1103/physreve.58.69.
- C. E. Tarnita, T. Antal, H. Ohtsuki, and M. A. Nowak. Evolutionary dynamics in set structured populations. *Proceedings of the National Academy of Sciences*, 106(21):8601–8604, May 2009. doi: 10.1073/pnas.0903019106.
- C. Taylor, D. Fudenberg, A. Sasaki, and M. A. Nowak. Evolutionary game dynamics in finite populations. *Bulletin of Mathematical Biology*, 66(6):1621–1644, Nov 2004. doi: 10.1016/j.bulm.2004.03.004.
- P. D. Taylor and L. B. Jonker. Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, 40(1-2):145–156, jul 1978. doi: 10.1016/0025-5564(78)90077-9.
- P. D. Taylor, A. J. Irwin, and T. Day. Inclusive fitness in finite deme-structured and stepping-stone populations. *Selection*, 1(1):153–164, Jan 2001. doi: 10.1556/select.1.2000.1-3.15.
- P. D. Taylor, T. Day, and G. Wild. Evolution of cooperation in a finite homogeneous graph. *Nature*, 447(7143):469–472, May 2007. doi: 10.1038/nature05784.
- A. Traulsen, J. M. Pacheco, and M. A. Nowak. Pairwise comparison and selection temperature in evolutionary game dynamics. *Journal of Theoretical Biology*, 246(3):522–529, Jun 2007. doi: 10.1016/j.jtbi.2007.01.002.
- A. Traulsen, C. Hauert, H. De Silva, M. A. Nowak, and K. Sigmund. Exploration dynamics in evolutionary games. *Proceedings of the National Academy of Sciences*, 106(3):709–712, Jan 2009. doi: 10.1073/pnas.

0808450106.

- Nicholas Wage. Evolutionary games on structured populations under weak selection. Harvard senior thesis, 2010.
- J. Y. Wakano and Y. Iwasa. Evolutionary branching in a finite population: Deterministic branching vs. stochastic branching. *Genetics*, 193(1):229–241, Oct 2012. doi: 10.1534/genetics.112.144980.
- J. Wakeley and T. Takahashi. The many-demes limit for selection and drift in a subdivided population. *Theoretical Population Biology*, 66(2):83–91, Sep 2004. doi: 10.1016/j.tpb.2004.04.005.
- L. Wardil and C. Hauert. Origin and structure of dynamic cooperative networks. *Scientific Reports*, 4, Jul 2014. doi: 10.1038/srep05725.